UNCLASSIFIED

AD 426189

DEFENSE DOCUMENTATION CENTER

FOR

SCIENTIFIC AND TECHNICAL INFORMATION

CAMERON STATION, ALEXANDRIA, VIRGINIA

UNCLASSIFIED

NOTICE: When government or other drawings, specifications or other data are used for any purpose other than in connection with a definitely related government procurement operation, the U. S. Government thereby incurs no responsibility, nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

SSD-TDR-63-158

REPORT NO.
TDR-169(3230-11)TN-15

426189

CATALOGED BY DDC
AS AD No.

Flutter and Vibration Analysis by a Modal Method: Analytical Development and Computational Procedure

31 JULY 1963

Prepared by WILLIAM P. RODDEN and EDITH F. FARKAS
Aeromechanics Department
Aerodynamics and Propulsion Research Laboratory
Laboratories Division
and
HEATHER A. MALCOM
Computation and Data Processing Center

Prepared for COMMANDER SPACE SYSTEMS DIVISION
UNITED STATES AIR FORCE
Inglewood, California

LABORATORIES DIVISION • AEROSPACE CORPORATION
CONTRACT NO. AF 04(695)-169

DDC
JAN 6 1964

SSD-TDR-63-158

Report No.
TDR-169(3230-11)TN-15

FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE

Prepared by

William P. Rodden and Edith F. Farkas
Aeromechanics Department
Aerodynamics and Propulsion Research Laboratory
Laboratories Division

and

Heather A. Malcom
Computation and Data Processing Center

AEROSPACE CORPORATION
El Segundo, California

Contract No. AF 04(695)-169

31 July 1963

Prepared for

COMMANDER SPACE SYSTEMS DIVISION
UNITED STATES AIR FORCE
Inglewood, California

SSD-TDR-63-158

Report No.
TDR-169(3230-11)TN-15

FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE

Prepared by: William P. Rodden
William P. Rodden, Head
Dynamics Section
Aerodynamics and Propulsion
Research Laboratory

Edith F. Farkas
Edith F. Farkas, MTS
Dynamics Section
Aerodynamics and Propulsion
Research Laboratory

Heather A. Malcom
Heather A. Malcom, MTS
Computation and Data
Processing Center

Approved by:
J. G. Logan, Director
Aerodynamics and Propulsion
Research Laboratory

B A Troesch
B. A. Troesch
Department Head,
Computation and Data
Processing Center

AEROSPACE CORPORATION
El Segundo, California

ABSTRACT

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilizes a newly developed subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system.

The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

CONTENTS

SYMBOLS

A	Element of aerodynamic force matrix, lb
a	Element of normalized flexibility matrix, in/lb
a_F	Amplitude coefficient of cantilever free vibration modal series, in.
a_G	Amplitude coefficient of generalized centroidal coordinates
a_R	Generalized amplitude coefficient of rigid-body modal series
a_{RF}	Element of rigid component modal matrix
b_r	Reference semichord, ft
C_h	Element of oscillatory aerodynamic influence coefficient matrix
F	Number of control points at which the modes are specified
g	Structural damping coefficient
h_o	Control point deflection due to rigid-body motion, in.
h_1	Control point deflection, in.
K	Flexibility matrix normalizing constant
K_F	Element of generalized spring matrix
k	Element of cantilever spring matrix, lb/in.
k_r	Reference reduced frequency
M	Element of mass matrix, lb
$\overline{M}$	Element of complex mass matrix including aerodynamic effects, lb
m	Element of generalized mass matrix
$\overline{m}$	Element of sum of generalized mass and aerodynamic matrices
N	Number of assumed flexible modes

SYMBOLS (Continued)

Q	Element of generalized aerodynamic force matrix
R	Number of rigid-body degrees of freedom
s	Reference semispan, ft (i.e., span measured from root to tip)
V	Velocity, knots
W	Element of aerodynamic weighting matrix
x, y	Cartesian coordinates
ϵ	Error of series solution
ρ	Air density, slugs/ft^3
Ω	Eigenvalue, (cps)$^{-2}$
ω	Frequency, cps
( )'	Prime denotes use of free-free modes
[ ]	Square matrix
{ }	Column matrix
⌈ ⌋	Diagonal matrix
$[\ ]^T$	Transposed matrix
$[\ ]^{-1}$	Inverse matrix

SECTION I

FORMULATION OF PROBLEM

A. Introduction

The use of structural influence coefficients in a collocation method of vibration analysis and of both structural and aerodynamic influence coefficients in a collocation flutter analysis (both of which are reviewed in Ref. 1) provides a number of advantages in terms of accuracy and convenience. However, the computation time in handling very large matrices becomes prohibitive, while computer capacity also may limit the analysis of extremely large systems, and a series or modal solution must be used. A modal formulation of the flutter problem using aerodynamic influence coefficients has been presented in Ref. 2. The present report reviews the derivations in Ref. 2 and extends its computational aspects, particularly in regard to the simplifications that accompany an option for a modal vibration analysis and an option to vary the density as well as the reduced velocity in the flutter analysis. The present program also utilizes a recently developed subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and its requirement for modes and frequencies is consistent with those generated by the vibration option of Ref. 1.

B. Sign Convention

The NASA body axis system with the x, y, and z axes directed forward, starboard, and downward, respectively, is recommended for consistency with the formulation of the static aeroelastic problems in Ref. 3. However, the usual flutter convention with the x, y, and z axes directed aft, starboard, and downward, respectively, may be used instead. In either case, the vertical normal force and deflection are positive downward.

C. Derivation of Equations

It is convenient to present two derivations of the modal formulation of the flutter and vibration problems according to whether cantilever or

free-free modes are to be utilized. The design analysis normally will make use of cantilever modes, but often the final analysis will be able to take advantage of the free-free modes available from ground vibration measurements. The two derivations are only slightly different; nevertheless it is useful to separate them.

The modal formulation may be derived equally well from an assumed series for the deflection modes and either Lagrange's equations for the dynamic system or the collocation (lumped parameter) equations of motion and Galerkin's method. Since the collocation equations of motion have been developed in Ref. 1, it is convenient to choose the latter approach. It is further convenient to summarize the significant features in the derivation of Ref. 1 to provide the basis for the present derivation. In Ref. 1, a single type of coordinate, viz., the deflection h, has been chosen as an adequate measure of both the deformation and the free-stream disturbance, not only for simplicity in the resulting equations but also because deflections have a more general meaning on a cambered vehicle and deflection influence coefficients are more readily obtained from a structural analysis than slope (or twist) influence coefficients. The resulting deformation matrix equation becomes

$$\{h_1\} - \{h_o\} = K[a](\{F_i\} + \{F_a\}) \tag{1}$$

where $\{h_1\}$ is the set of components of the absolute deflections of the control points, $\{h_o\}$ is the set of components of the deflections of the control points due to the rigid-body motion of some reference point, $[a]$ is the set of structural influence coefficients (SICs, or flexibility matrix) for the system <u>cantilevered</u> from (or otherwise restrained at) the reference point, $\{F_i\}$ is the set of inertial force components integrated throughout the region adjacent to each control point, $\{F_a\}$ is the set of aerodynamic force components integrated over the vehicle surface adjacent to each control point, and the scaler K has been introduced as a factor to the SICs for convenience in investigating variations in stiffness levels.

The inertial forces may be written in terms of a mass matrix [M] and the control point accelerations.

$$\{F_i\} = -(1/386)[M]\{\ddot{h}_1\} \tag{2}$$

where the diagonal elements of the mass matrix are found from integrating the structural mass density throughout the region adjacent to the control points. (N. B. The mass matrix need not be diagonal, and, in general, will not be so if the elements must be derived from a set of weight data previously lumped at a system of control points different from those required in the aeroelastic analysis. The use of a coupled mass matrix permits simulation of given inertial characteristics at a set of control points frequently dictated by more difficult aerodynamic considerations. A special technique for deriving the elements of a coupled mass matrix corresponding to a specified set of control points from a discrete set of given weight data is suggested in Appendix B of Ref. 1.)

The aerodynamic forces may be written in terms of a set of aerodynamic influence coefficients (AICs) for <u>oscillatory</u> motion and the control point deflections as

$$\{F_a\} = (4\pi^2/12)\, \rho\omega^2 b_r^2 s\, [W][C_h]\{h_1\} \tag{3}$$

where $[C_h]$ is the theoretically derived dimensionless (complex) matrix of oscillatory AICs, ω is the frequency of the assumed harmonic motion, ρ is the atmospheric density, b_r is the reference semichord, s is the reference span, and [W] is an empirically derived weighting matrix for modification of the theoretical AICs. A method for obtaining the elements of the weighting matrix has been suggested in Ref. 4.

The sum of the force components may now be written from Eqs. (2) and (3) for the case of harmonic motion.

$$\{F_i\} + \{F_a\} = (4\pi^2\omega^2/386)([M] + 32.174\,\rho b_r^2 s[W][C_h])\{h_1\} \tag{4a}$$

$$= (4\pi^2\omega^2/386)([M] + [A])\{h_1\} \tag{4b}$$

$$= (4\pi^2\omega^2/386)[\overline{M}]\{h_1\} \tag{4c}$$

We next discuss the manner of inclusion of the rigid-body degrees of freedom in Eq. (1). The matrix $\{h_o\}$ has been defined as the set of components of the deflections of the control points due to the rigid-body motion of the reference point. Each component of the control point deflections h_o is linearly related to the rigid-body translations and rotations provided the rotations are small. Therefore, we may define a rigid-body modal matrix $[h_R]$ as the transformation

$$\{h_o\} = [h_R]\{a_R\} \tag{5}$$

where $\{a_R\}$ is the set of amplitudes of rigid-body translations and rotations of the reference point. As an example, if we consider symmetrical vertical motion, $[h_R]$ is composed of two columns: The first is a unit column corresponding to the plunging mode; the second consists of the x-coordinate of each control point corresponding to the pitching mode; $\{a_R\}$ is composed of two elements: The first is the plunging displacement z_o; the second is the pitching angular displacement θ.

Before proceeding in the derivation, we should review the format of the various matrices in the case of a multiple flexible component system. As an example, we consider a symmetrical flutter analysis of an aircraft whose wing, aft fuselage, and tail are flexible and whose forward fuselage may be assumed to be rigid. We assume that the reference point (the cantilever

point) can be located in the vehicle such that its various components are independent. If we choose a point at the intersection of the wing and fuselage, then the wing is independent of the aft fuselage-tail combination, but the tail and aft fuselage must be considered together. The motion of the rigid forward fuselage is determined by the motion of the reference point, and the forward fuselage will not enter into any flexible considerations but only into the free-free boundary conditions. From the foregoing, it is seen that the various matrices will appear in partitioned form. If we denote the wing and aft fuselage-tail system by the subscripts 1 and 2, respectively, then the flexibility matrix appears as

$$[a] = \left[\begin{array}{c|c} a_1 & 0 \\ \hline 0 & a_2 \end{array}\right] \tag{6}$$

the mass matrix as

$$[M] = \left[\begin{array}{c|c} M_1 & 0 \\ \hline 0 & M_2 \end{array}\right] \tag{7}$$

the weighting matrix as

$$[W] = \left[\begin{array}{c|c} W_1 & 0 \\ \hline 0 & W_2 \end{array}\right] \tag{8}$$

the AICs as

$$[C_h] = \left[\begin{array}{c|c} C_{h1} & 0 \\ \hline 0 & C_{h2} \end{array}\right] \tag{9}$$

and the rigid body modal matrix as

$$[h_R] = \begin{bmatrix} h_{R1} \\ \hdashline h_{R2} \end{bmatrix} \tag{10}$$

Two requirements should be emphasized with regard to the AICs. The first concerns the proper inclusion of the reference geometry associated with the nondimensional AICs. The dimensional form of Eq. (9) may be written

$$b_r^2 s[C_h] = \left[\begin{array}{c:c} b_1^2 s_1 C_{h1} & 0 \\ \hdashline 0 & b_2^2 s_2 C_{h2} \end{array}\right] \tag{11}$$

where b_r and s are the reference semichord and span of the composite system, b_1 and s_1 are the reference geometry for the first component, and b_2 and s_2 are the reference geometry for the second component. The second requirement is that the AICs for each component must be determined for the same "dimensional" reduced velocity V/ω. If the reference reduced velocity is

$$1/k_r = V/b_r\omega \tag{12}$$

then the reduced velocity for the first component must be

$$1/k_1 = (1/k_r)(b_r/b_1) \tag{13}$$

and for the second component

$$1/k_2 = (1/k_r)(b_r/b_2) \tag{14}$$

Both of these requirements can be met in formulating the composite AICs by choosing the same reference geometry in determining the AICs for each component.

The rigid-body modal matrix provides the basis for a general statement of the boundary conditions for the free-free flutter of the composite system. The boundary conditions for harmonic motion may be written as

$$[h_R]^T[\overline{M}]\{h_1\} + [\Delta\overline{m}]\{a_R\} = \{0\} \tag{15}$$

where $[\Delta\overline{m}]$ is an incremental generalized mass matrix, including aerodynamic effects, of any rigid component of the system attached to the reference point (e. g. , the forward fuselage that was assumed to be rigid in the foregoing example), and not considered in the formulation of the flexible component mass and aerodynamic matrices. The form of the matrix $[\Delta\overline{m}]$ may be illustrated by the previous example with the rigid forward fuselage again in symmetrical motion. We may write

$$[\Delta\overline{m}] = [\Delta m] + [\Delta Q] \tag{16}$$

where the generalized rigid component mass matrix of the forward fuselage is

$$[\Delta m] = \begin{bmatrix} M_o & S_o \\ S_o & I_{yo} \end{bmatrix} \tag{17}$$

in which M_o, S_o, and I_{yo} are the mass, static unbalance about the reference point, and pitching moment of inertia about the reference point, respectively, of the forward fuselage, and the generalized aerodynamic forces on the forward fuselage (if not negligible) are found from

$$[\Delta Q] = 32.174\,\rho b_o^2 s_o [h_{Ro}]^T [C_{ho}][h_{Ro}] \tag{18}$$

where $[h_{Ro}]$ is the rigid-body modal matrix, $[C_{ho}]$ is the set of AICs, and b_o and s_o are the reference geometry for the forward fuselage. Again the AICs must be found for the reduced velocity of the composite system.

The collocation equation for flutter follows by substituting Eq. (4c) into Eq. (1) and adding the structural damping factor $1/(1 + ig)$ to the flexibility matrix to provide the artificial structural damping necessary to sustain the assumed harmonic motion of the flutter system

$$\{h_1\} - \{h_o\} = (4\pi^2 K\omega^2/386\ (1 + ig))[a][\overline{M}]\{h_1\} \tag{19}$$

The rigid-body degrees of freedom may be eliminated by use of Eqs. (5) and (15). For the purpose of seeking series solutions to Eq. (19), it is convenient to introduce a cantilever spring matrix defined by

$$[k] = (386/K)[a]^{-1} \tag{20}$$

and to rewrite the flutter equation as

$$\Omega[k](\{h_1\} - \{h_o\}) = 4\pi^2[\overline{M}]\{h_1\} \tag{21}$$

where the eigenvalue $\Omega = (1 + ig)/\omega^2$.

This review of the derivation of the collocation equation and of the formats of the various matrices places us in a position to consider the two modal formulations of the flutter and vibration problems. We consider first the choice of a series of cantilever modes, and then the choice of a series of free-free modes.

Case I: Cantilever Modes

Let us denote by N the number of flexible modes given, by F the number of control points at which the modes are specified, and by R the number of rigid-body degrees of freedom. If we select a series of cantilever modes (which satisfy the requirement of the Galerkin method that all boundary conditions be met) for the solution of Eq. (21) then the series may be written

$$\{h_1\} - \{h_o\} = [h_F]\{a_F\} \tag{22}$$

where $[h_F]$ is the matrix of given cantilever modes and the $\{a_F\}$ are the modal amplitudes. The matrix $[h_F]$ appears in a partitioned form (size F × N) in which each partition corresponds to a component of the composite system

$$[h_F] = \left[\begin{array}{c:c} h_{F1} & 0 \\ \hdashline 0 & h_{F2} \end{array}\right] \tag{23}$$

Substituting Eqs. (5) and (22) into Eq. (15) yields

$$[\overline{m}_{RF}]\{a_F\} + [\overline{m}_{RR}]\{a_R\} = 0 \tag{24}$$

where

$$[\overline{m}_{RF}] = [m_{RF}] + [Q_{RF}] \tag{25a}$$

$$= [h_R]^T[M][h_F] + [h_R]^T[A][h_F] \tag{25b}$$

$$[\overline{m}_{RR}] = [m_{RR}] + [Q_{RR}] \tag{26a}$$

$$= ([h_R]^T[M][h_R] + [\Delta m]) + ([h_R]^T[A][h_R] + [\Delta Q]) \tag{26b}$$

and from which

$$\{a_R\} = -[\overline{m}_{RR}]^{-1}[\overline{m}_{RF}]\{a_F\} \tag{27}$$

The flutter equation, Eq. (21), now becomes

$$\Omega[k][h_F]\{a_F\} = 4\pi^2[\overline{M}]([h_F]\{a_F\} + [h_R]\{a_R\}) + \{\epsilon\} \tag{28}$$

where $\{\epsilon\}$ is the error in the series solution. Applying the Galerkin principle ($[h_F]^T\{\epsilon\} = 0$) yields

$$\Omega[K_F]\{a_F\} = 4\pi^2([\overline{m}_{FF}]\{a_F\} + [\overline{m}_{FR}]\{a_R\}) \tag{29}$$

where

$$[K_F] = [h_F]^T[k][h_F] \tag{30}$$

$$[\overline{m}_{FF}] = [m_{FF}] + [Q_{FF}] \tag{31a}$$

$$= [h_F]^T[M][h_F] + [h_F]^T[A][h_F] \tag{31b}$$

$$[\overline{m}_{FR}] = [m_{FR}] + [Q_{FR}] \tag{32a}$$

$$= [m_{RF}]^T + [h_F]^T[A][h_R] \tag{32b}$$

If we eliminate the rigid-body degrees of freedom by substituting Eq. (27) into Eq. (29), we obtain the final form of the cantilever modal formulation

$$\Omega[K_F]\{a_F\} = 4\pi^2([\overline{m}_{FF}] - [\overline{m}_{FR}][\overline{m}_{RR}]^{-1}[\overline{m}_{RF}])\{a_F\} \tag{33}$$

or in the canonical form

$$\Omega\{a_F\} = \lceil K_F/4\pi^2\rfloor^{-1}([\overline{m}_{FF}] - [\overline{m}_{FR}][\overline{m}_{RR}]^{-1}[\overline{m}_{RF}])\{a_F\} \tag{34}$$

We note that the generalized spring matrix is diagonal because our formulation is not statically coupled (it is dynamically coupled instead). The generalized spring matrix may be found in terms of the generalized free vibration mass matrix and the natural (normal or uncoupled) frequencies

$$\lceil K_F/4\pi^2\rfloor = \lceil \text{diagonal elements of } [m_{FF}]\rfloor \lceil \omega_F^2 \rfloor \tag{35}$$

The flutter equation, Eq. (34), can be solved by the power method for the eigenvalues, $\Omega = \Omega_R + i\Omega_I$, from which we find the frequency

$$\omega = 1/\sqrt{\Omega_R} \qquad \text{(cps)} \tag{36}$$

the required structural damping

$$g = \Omega_I/\Omega_R \tag{37}$$

and, since the aerodynamic matrix required the assumption of a reduced velocity, $1/k_r$, for its formulation, the velocity

$$V = 0.5921(2\pi\omega b_r)(1/k_r) \qquad \text{(knots)} \tag{38}$$

The power method as developed for non-Hermitian matrices with close roots is discussed in Appendix A of Ref. 1. The flutter mode is found from the originally assumed series

$$\{h_1\} = [h_F]\{a_F\} + [h_R]\{a_R\} \tag{39a}$$

$$= ([h_F] - [h_R][\overline{m}_{RR}]^{-1}[\overline{m}_{RF}])\{a_F\} \tag{39b}$$

From a series of solutions of Eq. (34) for various reduced velocities the conventional required damping versus velocity stability curve can be constructed for a specific altitude and the flutter point is determined as the velocity for which the required damping and actual damping are equal. An alternative approach to the flutter analysis is based on a single representative reduced velocity and a series of solutions of Eq. (34), carried out for various densities. The density at which the required damping and actual damping are equal may be used to find a stiffness-altitude similarity parameter for flutter from which the flutter stability may be determined. However, at present, the validity of this latter approach requires further investigation, particularly the sensitivity of the results to the choice of representative reduced velocity.

The generalized mass corresponding to each free vibration mode is also of interest in various modal analyses of flying qualities, stability and control characteristics, or transient response of flexible vehicles. If Eq. (34) is solved for the free vibration modes (by deleting the aerodynamic terms), then the n'th generalized mass is given by*

$$m^{(n)} = \{h_1^{(n)}\}^T[M]\{h_1^{(n)}\} + \{a_R^{(n)}\}^T[\Delta m]\{a_R^{(n)}\} \tag{40a}$$

*N. B. When the vibration modes $\{h_1^{(n)}\}$ are normalized on their maximum elements, the modal amplitudes $\{a_F^{(n)}\}$ cannot be normalized but must be scaled accordingly for the calculation of the generalized masses.

$$m^{(n)} = \{a_F^{(n)}\}^T[m_{FF}]\{a_F^{(n)}\} + \{a_R^{(n)}\}^T[m_{RR}]\{a_R^{(n)}\}$$

$$+ \{a_R^{(n)}\}^T[m_{RF}]\{a_F^{(n)}\} + (\{a_R^{(n)}\}^T[m_{RF}]\{a_F^{(n)}\})^T \tag{40b}$$

$$= \{a_F^{(n)}\}^T ([m_{FF}]\{a_F^{(n)}\} + [m_{RF}]^T\{a_R^{(n)}\}) \tag{40c}$$

where $\{a_F^{(n)}\}$ is the n'th set of cantilever modal amplitudes and the corresponding rigid component mode is found from Eq. (27) (again without the aerodynamic terms).

$$\{a_R^{(n)}\} = -[m_{RR}]^{-1}[m_{RF}]\{a_F^{(n)}\} \tag{41}$$

<u>Case II: Free-Free Modes</u>

When free-free vibration data are available for a flutter analysis, it is necessary to write two series expressions that provide the basis for the modal solution.

$$\{h_1\} = [h'_F]\{a'_F\} + [h_R]\{a_G\} \tag{42}$$

$$\{a_R\} = [a_{RF}]\{a'_F\} + \{a_G\} \tag{43}$$

where the prime denotes the free-free condition, $\{a_G\}$ is the matrix of centroidal generalized coordinates, and $[a_{RF}]$ is the rigid component modal matrix. The rigid component modal matrix may be found from the boundary conditions for the free vibration. However, it is desirable to permit the consideration of some cantilever modes in addition to the orthogonal free-free modes. Hence, if we write

$$[h'_F] = [h'_{Fo} \mid h'_{Fc}] \tag{44}$$

then

$$[a_{RF}] = [a_{RFo} \mid 0] \tag{45}$$

where the subscripts o and c denote orthogonal and cantilever, respectively. The orthogonal partition follows from the free vibration boundary condition and is given by

$$[a_{RFo}] = -[\Delta m]^{-1}[h_R]^T[M][h'_{Fo}] \tag{46}$$

Making the series substitutions, Eqs. (42) and (43), into the boundary condition, Eq. (15), yields

$$[h_R]^T[\overline{M}]([h'_F]\{a'_F\} + [h_R]\{a_G\}) + [\Delta\overline{m}]([a_{RF}]\{a'_F\} + \{a_G\}) = 0 \tag{47}$$

from which

$$\{a_G\} = -[\overline{m}_{RR}]^{-1}[\overline{m}'_{RF}]\{a'_F\} \tag{48}$$

where

$$[\overline{m}'_{RF}] = [h_R]^T[\overline{M}][h'_F] + [\Delta\overline{m}][a_{RF}] \tag{49}$$

In order to apply the Galerkin principle when free-free modes are used in the series solution, it is necessary to obtain the equation of motion of the rigid

component. If the flexible system mass term is eliminated between Eqs. (15) and (21), then the rigid component equation of motion is obtained.

$$-\Omega[h_R]^T[k](\{h_1\} - \{h_o\}) = 4\pi^2[\Delta\overline{m}]\{a_R\} \tag{50}$$

Substituting the series solutions, Eqs. (42) and (43), into Eq. (50) yields

$$\Omega[h_R]^T[k][h''_F]\{a'_F\} = 4\pi^2[\Delta\overline{m}]([a_{RF}]\{a'_F\} + \{a_G\}) + \{\epsilon_R\} \tag{51}$$

where $\{\epsilon_R\}$ is the error in Eq. (50) due to the series solution and

$$[h''_F] = [h'_F] - [h_R][a_{RF}] \tag{52}$$

Substituting the series solutions into the flexible system equation of motion, Eq. (21), yields

$$\Omega[k][h''_F]\{a'_F\} = 4\pi^2[\overline{M}]([h'_F]\{a'_F\} + [h_R]\{a_G\}) + \{\epsilon_F\} \tag{53}$$

where $\{\epsilon_F\}$ is the error in Eq. (21) due to the series solution. Applying the Galerkin principle (in this case, $[a_{RF}]^T\{\epsilon_R\} + [h'_F]^T\{\epsilon_F\} = 0$) yields the generalized equation of motion

$$\Omega[K'_F]\{a'_F\} = 4\pi^2([\overline{m}'_{FF}]\{a'_F\} + [\overline{m}'_{FR}]\{a_G\}) \tag{54}$$

where

$$[K'_F] = [h''_F]^T[k][h''_F] \tag{55}$$

$$[\overline{m}'_{FF}] = [m'_{FF}] + [Q'_{FF}] \tag{56a}$$

$$= ([h'_F]^T[M][h'_F] + [a_{RF}]^T[\Delta m][a_{RF}])$$

$$+ ([h'_F]^T[A][h'_F] + [a_{RF}]^T[\Delta Q][a_{RF}]) \tag{56b}$$

$$[\overline{m}'_{FR}] = [h'_F]^T[\overline{M}][h_R] + [a_{RF}]^T[\Delta\overline{m}] \tag{57}$$

If we eliminate the centroidal degrees of freedom by substituting Eq. (48) into Eq. (54), we obtain the final form of the free-free modal formation

$$\Omega\lceil K'_F \rfloor\{a'_F\} = 4\pi^2([\overline{m}'_{FF}] - [\overline{m}'_{FR}][\overline{m}_{RR}]^{-1}[\overline{m}'_{RF}])\{a'_F\} \tag{58}$$

or in the canonical form

$$\Omega\{a'_F\} = \lceil K'_F/4\pi^2 \rfloor^{-1}([\overline{m}'_{FF}] - [\overline{m}'_{FR}][\overline{m}_{RR}]^{-1}[\overline{m}'_{RF}])\{a'_F\} \tag{59}$$

We again note that the diagonal generalized spring matrix is found from the generalized mass matrix and natural frequencies

$$\lceil K'_F/4\pi^2 \rfloor = \lceil \text{diagonal elements of } [m'_{FF}] \rfloor \lceil \omega'^2_F \rfloor \tag{60}$$

The solution of the flutter equation by iteration has been discussed in the previous case. The flutter mode follows from the assumed series. Substituting Eq. (48) into Eq. (42) yields

$$\{h_1\} = ([h'_F] - [h_R][\overline{m}_{RR}]^{-1}[\overline{m}'_{RF}])\{a'_F\} \tag{61}$$

The n'th generalized mass follows from the solution of Eq. (59) with the aerodynamic terms deleted, and Eqs. (40a), (42), and (43).*

$$m^{(n)} = \{a_F'^{(n)}\}^T [m'_{FF}]\{a_F'^{(n)}\} + \{a_G^{(n)}\}^T [m_{RR}]\{a_G^{(n)}\}$$

$$+ \{a_G^{(n)}\}^T [m'_{RF}]\{a_F'^{(n)}\} + (\{a_G^{(n)}\}^T [m'_{RF}]\{a_F'^{(n)}\})^T \tag{62}$$

$$= \{a_F'^{(n)}\}^T ([m'_{FF}]\{a_F'^{(n)}\} + [m'_{RF}]^T \{a_G^{(n)}\}) \tag{62a}$$

where

$$\{a_G^{(n)}\} = -[m_{RR}]^{-1}[m'_{RF}]\{a'_F\} \tag{63}$$

D. References

1. W. P. Rodden, E. F. Farkas and H. A. Malcom. "Flutter and Vibration Analysis by a Collocation Method: Analytical Development and Computational Procedure." Aerospace Corporation Report No. TDR-169(3230-11)TN-14, 31 July 1963.

2. W. P. Rodden, E. F. Farkas, P. E. Williams, and F. C. Slack. "Flutter Analysis by a Modal Method Using Aerodynamic Influence Coefficients: Analytical Development and Procedure for the IBM 7090 Computer." Norair Division, Northrop Corporation Report NOR-61-54, 14 April 1961.

*N. B. When the vibration modes $\{h_1^{(n)}\}$ are normalized on their maximum elements, the modal amplitudes $\{a_F'^{(n)}\}$ cannot be normalized but must be scaled accordingly for the calculation of the generalized masses.

3. W. P. Rodden, E. F. Farkas, and H. A. Malcom. "Quasi-Static Aero-Thermo-Elastic Analysis: Analytical Development and Computational Procedure." Aerospace Corporation Report No. TDR-169(3230-11)TN-8, 1 March 1963.

4. W. P. Rodden. "An Empirical Weighting Matrix for Use with Aerodynamic Influence Coefficients in Aeroelastic Analyses." Northrop Corporation Report NOR-59-320, 1 April 1959.

5. R. L. Bisplinghoff, H. Ashley, and R. L. Halfman. Aeroelasticity. Reading: Addison-Wesley Publishing Company, Inc., 1955.

SECTION II

GENERAL DESCRIPTION OF INPUT

A. Units

The dimensional data required for each component consist of the mass matrix in pounds, the free vibration frequencies in cycles per second, and the reference semichords and semispans in feet. The free vibration modes and the aerodynamic influence coefficients are dimensionless. Compatibility is required between the units of the rigid-body modal matrices and the generalized rigid-body mass matrix; for example, if S_o and I_{yo} are in the pound-foot system, then the x-coordinates, which correspond to the pitching mode, must be measured in feet, whereas if the generalized masses are given in the pound-inch system, the x-coordinates must be measured in inches. The density is required in slugs per cubic foot.

B. Classes of Data and Example Problems

Five classes of data must be provided: mass, aerodynamic influence coefficients (AICs) and their weighting matrices, free vibration modes (cantilever and/or free-free) and corresponding frequencies, rigid-body modes, and the generalized masses for any rigid component in the system. (The cantilever case does not require the rigid-body modes or the generalized masses).

Two example problems will be considered. Each problem will be a symmetrical flutter analysis of the jet transport wing treated as an example throughout Ref. 5. In the first analysis, we will use cantilever vibration modes and frequencies, and in the second we use free-free symmetric modes and frequencies such as might be obtained from a ground vibration survey. In each problem, we assume that four modes will be sufficient to obtain convergence in the flutter solution.

The four flexible cantilever modes (calculated by the collocation method of Ref. 1) are

$$[h_F] =$$

$$\begin{bmatrix} 0.045185 & -0.163556 & -0.219686 & 0.291415 \\ 0.042502 & 0.544990 & -0.243887 & 0.279638 \\ 0.148256 & -0.279631 & -0.205463 & 0.306804 \\ 0.143727 & 1.0 & -0.362884 & 0.353608 \\ 0.379411 & -0.226740 & -0.371824 & -0.559460 \\ 0.376733 & 0.946563 & -0.428613 & -0.490161 \\ 0.692995 & -0.138889 & 0.203824 & -0.146405 \\ 0.692655 & 0.887880 & 0.163356 & -0.020285 \\ 0.998969 & -0.039737 & 1.0 & 0.837729 \\ 1.0 & 0.822370 & 0.981915 & 1.0 \end{bmatrix}$$

The corresponding frequencies are

$$\begin{aligned} \omega_1 &= 2.036802 \text{ cps} \\ \omega_2 &= 3.552589 \text{ cps} \\ \omega_3 &= 7.280460 \text{ cps} \\ \omega_4 &= 11.698543 \text{ cps} \end{aligned}$$

The four free-free symmetric modes (also found by the method of Ref. 1) are

$$[h_F] =$$

$$\begin{bmatrix} -0.129721 & -0.091809 & -0.035610 & 0.305979 \\ -0.131332 & 0.568102 & -0.047615 & 0.287803 \\ -0.017537 & -0.221627 & -0.162942 & 0.442843 \\ -0.015998 & 1.0 & -0.189775 & 0.503783 \\ 0.247806 & -0.215371 & -0.449387 & -0.561462 \\ 0.250893 & 0.905541 & -0.477740 & -0.443473 \\ 0.622665 & -0.205131 & 0.106948 & -0.225098 \\ 0.627940 & 0.776147 & 0.097215 & -0.006219 \\ 0.993742 & -0.186900 & 0.984408 & 0.741022 \\ 1.0 & 0.636687 & 1.0 & 1.0 \end{bmatrix}$$

and the corresponding frequencies are

$$\begin{aligned} \omega_1 &= 2.442962 \\ \omega_2 &= 3.603113 \\ \omega_3 &= 8.553045 \\ \omega_4 &= 12.707073 \end{aligned}$$

The symmetrical rigid component (one-half of the fuselage) generalized mass matrix is given below in the pound-inch system.

$$[\Delta m] = \begin{bmatrix} M_o & S_o \\ S_o & I_{yo} \end{bmatrix} = \begin{bmatrix} 17,400 & 1,370,250 \\ 1,370,250 & 4,457,907,200 \end{bmatrix}$$

(In the antisymmetric case, the rigid component generalized mass matrix is $[\Delta m] = [I_{xo}]$; in the composite symmetric-antisymmetric case, the generalized mass matrix is

$$[\Delta m] = \begin{bmatrix} M_o & S_o & 0 \\ S_o & I_{yo} & 0 \\ 0 & 0 & I_{xo} \end{bmatrix} \quad .)$$

The mass and aerodynamic matrices for the wing and the aerodynamic matrix for the fuselage are shown in the example problem printed output (pages 47-73) and will notbe repeated here. The aerodynamic weighting matrix is taken as unity, and no input is required.

In addition to the flexible modes, the symmetrical analysis requires the two rigid-body modes for the degrees of freedom of plunging and pitching. The rigid-body modal matrix therefore consists of two columns: a unit column and a column of the x-coordinate of each control point. The rigid-body modal matrix for the wing is

$$[h_R] = [1 \; x] = \begin{bmatrix} 1 & 20.25 \\ 1 & -81.00 \\ 1 & 17.85 \\ 1 & -71.40 \\ 1 & 15.80 \\ 1 & -63.20 \\ 1 & 13.30 \\ 1 & -53.20 \\ 1 & 11.05 \\ 1 & -44.20 \end{bmatrix}$$

The rigid-body modal matrix for the fuselage is

$$[h_{Ro}] = [1 \ x] \begin{bmatrix} 1 & -373.30 \\ 1 & -248.30 \\ 1 & -123.30 \\ 1 & +1.70 \\ 1 & +126.70 \end{bmatrix}$$

Note that the above matrix is used in computing the incremental complex generalized mass which results from the aerodynamic loads on one-half of the fuselage and, therefore, the x-coordinates must be given in the proper order for the control points used in computing the fuselage AICs. (In this problem the fuselage AICs are hypothetical but the x-coordinates are given in the order necessary for the slender-body theory AICs referred to later (page 39) in this report.)

The reference geometry for the wing is $b_{rw} = 5.468$ ft and $s_w = 37.917$ ft. The reference geometry for the fuselage is $b_{ro} = 5.468$ ft and $s_o = 18.9585$ ft. (We have assumed that the wing reference geometry was used to nondimensionalize the fuselage AIC matrix, and since we need only one-half of the fuselage aerodynamic forces they can be obtained by setting $s_o = (1/2)s_w$.) The flutter analyses are carried out for the reduced velocity $1/k_r = 16.67$ using the reference semichord for the system $b_r = 5.468$ ft. Both analyses are carried out for sea level; i.e., $\rho = 0.002378$ slugs per cubic foot.

The results for each of the two problems are shown in the example printed output, Section IV, Part A. The final results are shown also in the cases of 6-, 8-, and 10-mode solutions to demonstrate the convergence of the four-mode solution. We note that the four-mode convergence is very poor, emphasizing the importance of checking convergence in any modal solution. We also note that the 10-mode solution agrees exactly with the collocation solution shown in Ref. 1 as is to be expected for a system with 10 degrees

of freedom. A final point of interest is to note that the use of free-free input modes does not provide any better convergence than the cantilever modes.

C. Program Restrictions and Options

1. The maximum number of vibration modes that can be input for the system is 40.

2. The maximum number of cantilever vibration modes is 40 whether the system consists of one or more components.

3. The maximum number of free-free vibration modes for the system is 40 minus the total number of cantilever modes to be used for the system. Since free-free modes represent the entire flexible system, a portion of each free-free mode must be input corresponding to each flexible component.

4. The number of flexible components is ≤ 20.

5. The number of values used in the reference reduced velocity ($1/k_r$) series is ≤ 20.

6. The maximum number of values used in a density series is ≤ 20.

7. The program provides for varying the densities with each $1/k_r$ or for using the same densities with all $1/k_r$'s.

8. The maximum number of output modes is 25; however, the number cannot exceed the number of input vibration modes.

9. The maximum number of control points for the rigid component or any flexible surface is 49.

10. The number of rigid-body modes is ≤ 6.

11. It is possible to reserve a partition in the upper left-hand corner of any flexible component AIC matrix for control points whose aerodynamic forces may be neglected or found from some alternate theory to that used for the primary control points. This partition is termed the "external stores" region since external stores are an example of a source of additional control

points requiring such special consideration. The maximum number of control points that can be reserved on each surface for external stores is ≤ 48.

12. A weighting matrix is optional input for each surface. The order of this matrix must agree with the order of the AIC matrices for the particular surface.

13. Any number of complete sets (decks) of input data may be stacked and run in one machine pass.

SECTION III

DATA DECK SETUP

A. Loading Order

Input data decks are loaded behind column binary deck No. LD014A. The data decks are assembled using cards punched from keypunch forms and cards that are punched-card output from appropriate IBM programs. The data items in each deck have the following order with this exception: some of the items may be omitted if indicators used in the control cards (Items 2 and 4) specify their absence.

1. Title card.

2. Data deck general control card.

3. Data card for changes in matrix iteration tests.

4. Control card(s) for external stores and weighting matrices.

5. Reference semichord (b_r) and reference reduced velocities ($1/k_r$'s).

6. Reference semichords (b_{ri}) and reference semispans (s_i) for rigid and flexible components (surfaces).

7. Density series (if same densities are used for all $1/k_r$'s).

8. Generalized mass matrix ($[\Delta m]$) for rigid component.

9. Mass matrix ($[M]_i$) for each flexible surface.

10. Rigid-body modal matrix ($[h_{Ro}]$) for rigid component.

11. Rigid-body modal matrix ($[h_R]_i$) for each flexible surface.

12. Frequencies (ω_F) and vibration modal matrix ($[h_F]_i$) for each flexible surface.

13. Weighting matrices ($[W]_i$).

14. Aerodynamic input repeated for each $1/k_r$

a. Density series cards (if densities vary with each $1/k_r$),

b. AIC matrix ($[C_{ho}]$) for rigid component (if present),

c. AIC matrix ($[C_h]_i$) for each flexible surface.

B. <u>Input Data Description</u>

1. The title card may contain any alphanumeric characters desired in Columns 2 through 80. Column 1 should be blank.

2. Data deck general control card (FORMAT 18I4): The first 11 fields in this card contain, in the order defined, the following indicators:

NSUR: Number of flexible components (surfaces), ≤ 20.

NAERO: Number of reference reduced velocities, ≤ 20; NAERO = 0 is used for the vibration analysis.

NRIGID: Number of rigid-body motion modes to be input (= number of columns in $[h_R]$); NRIGID = 0 for the cantilever case.

NFUS: NFUS <u>must</u> = 1 if AICs ($[C_{ho}]_j$, j = 1, NAERO) are input for the rigid component. NFUS = 0 if $[C_{ho}]$ is not input.

NDENS: If NDENS = 0, the densities are to vary with each $1/k_r$ and are input as part of Item 14; if NDENS > 0, this number of densities must be input as Item 7 and each density value will be used for all $1/k_r$'s.

MODES: Number of output modes, ≤ 25.

NFREE: Number of free-free vibration modes to be input, $\leq$ (40 - NCANT), (see next indicator). Note that NFREE does not vary with the number of flexible components (surfaces) but each free-free mode will have NSUR partitions and each partition will consist of the control point deflections for one surface. NFREE = 0 if only cantilever modes are used.

NCANT: Total number of cantilever modes to be input for the system. $\leq$ (40 - NFREE)

NDELM: NDELM = 0 if no rigid component generalized mass matrix ([Δm]) is input; NDELM = 1 if [Δm] is input.

NPUNCH: This indicator is used to obtain a printout of the dynamic matrix ([U]) and to obtain the frequencies and modes in punched-card format. NPUNCH = 0 if no printout of [U] or punched output is desired; NPUNCH = 1 if only punched-card output is desired and NPUNCH = -1 will provide for the printout of [U] and the punched output. (The minus must be in Column 37 and the 1 in Column 40.)

NCON: This indicator provides for changing five program test numbers used in the matrix iteration subroutine. NCON = 0 if no changes are desired and NCON $\neq$ 0 if any of the tests are to be changed.

3. Data card for changes in matrix iteration tests (FORMAT 3E12.8 and 2I4): There are three test numbers and two control numbers associated with the iteration procedure (Ref. 1, Appendix A) that are present in the program but that may be changed by the program user. The numbers built into the program are given below. An explanation of its purpose follows each number.

EPSP = 0.5×10^{-6}; test for eigenvector convergence when the iteration procedure is approaching a single root.

EPDP = 0.5×10^{-7}; test for convergence when the iteration procedure is approaching a pair of close roots.

AITKEN = 0.9; if this test is met the program uses a procedure (known as Aitken's δ^2 method) to accelerate convergence.

NITRSP = 40; a maximum of 40 single-precision arithmetic iterations will be performed for each eigenvalue if its eigenvectors have not converged in a lesser number.

NITRDP = 100; if the eigenvectors for any one eigenvalue have not converged in NITRSP single-precision iterations the program will then use up to a maximum of 100 double-precision arithmetic iterations.

Altering any of the above numbers (NCON $\neq$ 0, Item 2) requires entering all five numbers on one card even if some of the numbers are to be the same as shown above. Input EPSP, EPDP, and AIKEN in the first three E12.8 fields on the card, the next two I4 fields (Columns 37 through 40 and 41 through 44) contain NITRSP and NITRDP, respectively. The EPSP and EPDP tests present in the program represent six-significant-figure accuracy although some loss in significance will occur as the higher modes are found; increasing these numbers would relax the requirements for accuracy and decrease the computing time. The AITKEN test must be less than but not too close to unity; an optimum convergence rate has been observed for values around 0.9. The NITRSP and NITRDP can be increased to the limit of the I4 fields, and either control may be decreased to zero. Errors in input data often create convergence problems, therefore all input should be carefully checked before an increase is made in the maximum number of iterations.

ITEMS 4, 5, 6, 7, 10, 13, and 14 DESCRIBED BELOW SHOULD BE OMITTED WHEN NAERO = 0 IS USED IN ITEM 2 (Vibration Analysis).

---

4. Control card(s) for external stores and weighting matrices (FORMAT 18I4): The consecutive fields in these cards contain $ISXT_1$, ISW_1, $ISXT_2$, ISW_2, . . . , $ISXT_{NSUR}$, ISW_{NSUR}. $ISXT_i$ = number of control points reserved for the external stores on surface i; if no external stores are present on surface i, enter zero for $ISXT_i$ or leave its field blank.

ISW_i = 0 (or blank field) if no weighting matrix is to be input for surface i, ISW_i = 1 if a weighting matrix will be input for surface i. Include only the number of cards needed for i = 1, NSUR.

5. Reference semichord, b_r and $1/k_r$ (sometimes denoted by $V/b_r\omega$) series (FORMAT 6E12.8): These reference parameters are used in computing the flutter velocities [Eq. (38)]. The b_r (feet) may be any value, but the proper relationship must exist between the $1/k_r$'s used for reference reduced velocities and the $1/k_{ri}$ used when computing the AIC matrices for each surface; that is, $1/k_{ri} = (1/k_r)(b_r/b_{ri})$ where b_{ri} is the reference semichord for surface i. To input these parameters enter b_r in Field 1 of the first card; Field 2 and consecutive fields in the first and successive cards (if NAERO > 5) contain the $1/k_r$'s, ≤ 20.

6. The b_{ri} and s_i series (FORMAT 6E12.8): A reference semichord (b_{ri}) and a reference semispan (s_i) must be input for the rigid component if NFUS = 1 and for each flexible component if NAERO > 0. When NFUS = 1 the consecutive fields in the first and successive (if NSUR > 2) card(s) contain in order b_{ro}, s_o, b_{r1}, s_1, . . . , b_{rNSUR}, and s_{NSUR}. When NFUS = 0, the consecutive fields in the necessary card(s) contain in order b_{r1}, s_1, b_{r2}, s_2, . . . , b_{rNSUR}, and s_{NSUR}.

7. Density series (FORMAT 6E12.8): Omit this input if NDEN = 0 in Item 2. If NDEN > 0, begin in Field 1 of the first card and enter NDEN densities, ≤ 20.

8. Rigid component generalized mass matrix $[\Delta m]$ (FORMAT 6E12.8): (Omit this matrix when NDELM = 0 in Item 2.) The size of the $[\Delta m]$ matrix for the rigid component(s) of the system must be compatible with the corresponding matrix for the flexible components given by $[h_R]^T[M][h_R]$; i.e., each element in $[\Delta m]$ is based upon the same rigid-body generalized coordinates as the respective element in the product matrix. If NDELM = 1, input the $[\Delta m]$ matrix (size NRIGID $\times$ NRIGID) by column with each column beginning on a new card.

9. Mass matrix, [M]: Although the [M] matrix defined in Section I, Part C, has NSUR × NSUR partitions, only the nonzero partitions are input; i. e., a separate mass matrix ($[M]_i$) is input for each surface. The sequence for considering the surfaces was established in Items 4 and 6. Repeat the following input for i = 1, NSUR.

a. Size control card $_i$ (FORMAT 18I4):

Field 1 contains $NSIZE_i$, the order of $[M]_i$ (external stores control points + surface control points, ≤ 49).

b. Control card(s) for omitting zeros in the coupled mass matrix $[M]_i$ (FORMAT 18I4):

Field 1 and consecutive fields in the first and successive cards (NSIZE > 9) contain in order LOW_1, $LHIGH_1$, LOW_2, $LHIGH_2$, . . . , LOW_{NSIZE}, $LHIGH_{NSIZE}$. LOW_j is the row number in which the first nonzero appears in Column j, and $LHIGH_j$ is the row number in which the last nonzero element appears in Column j. If only one nonzero element is in Column j, the row number in which it appears must be used for both LOW_j and $LHIGH_j$.

c. Elements in $[M]_i$ matrix (FORMAT 6E12.8): The elements are entered by column as shown below; the input for each column starts on a new card.

Column 1: elements LOW_1 through $LHIGH_1$.

Column 2: elements LOW_2 through $LHIGH_2$.

.
.
.

Column NSIZE: elements LOW_{NSIZE} through $LHIGH_{NSIZE}$.

Any zero elements appearing in rows between LOW_j and $LHIGH_j$ must be entered or their respective fields left blank. If all elements in a column are zero, one zero element must be entered for this column (a blank card

may be used) giving $LOW_j = LHIGH_j$ for this element in the above control cards. If external stores are present ($ISXT_i > 0$, Item 4) all store control points must be considered before the surface control points; i.e., the elements representing the external stores mass must occupy the upper left-hand corner of the mass matrix.

NOTE: One possible method for determining the [M] matrices is suggested in Appendix B of Ref. 1.

10. Rigid component modal matrix, $[h_{Ro}]$: (Omit this input when NFUS = 0, Item 2.) The rigid-body modes for the rigid component are input in order to compute the incremental generalized force matrix $[\Delta Q]$ that results from aerodynamic loads on the component. The number of rows in $[h_{Ro}]$ must be the same as the number of control points considered when computing the $[C_{ho}]$ matrices, the number of columns must agree with NRIGID, Item 2. These data are input in the following manner.

a. Size card (FORMAT 18I4):

Field 1 contains NROWS, number of rows in $[h_{Ro}]$.

b. Elements in $[h_{Ro}]$ (FORMAT 6E12.8): Tabulate the matrix elements by column with each column beginning on a new card.

11. Rigid-body modal matrix, $[h_R]$: (Omit this input if NRIGID = 0, Item 2.) If NRIGID > 0, input the $[h_R]$ matrix by partitions; i.e., input $[h_R]_i$ size ($NSIZE_i \times NRIGID$) for each surface. The $[h_R]_i$ matrix requires the following input, repeating the input for i = 1, NSUR.

a. Size control card $_i$ (FORMAT 18I4):

Field 1 contains $NSIZE_i$, the number of rows (≤49) in $[h_R]_i$ (must agree with order of $[M_i]$).

b. $[h_R]_i$ matrix elements (FORMAT 6E12.8): Enter the matrix elements by column with each column beginning on a new card.

12. Vibration modes ($[h_F]$ matrix) and frequencies: The $[h_F]$ matrix is also partitioned and a separate $[h_F]_i$ matrix is considered for each surface. The columns in $[h_F]_i$ are the free vibration modes (orthogonal free-free and/or cantilever) for surface i. The total number of modes (NFREE + NCANT) used for the system cannot exceed 40. The frequency (cps) associated with each mode must also be input.

Input the frequencies and modal column matrices ($[h_F]_i$) in the following manner: if NFREE = 0 repeat Parts a, b, and c (below) for i = 1, NSUR; if NFREE > 0 omit Part b for i = 2, NSUR.

a. Control card $_i$ (FORMAT 18I4):

Field 1 contains $NSIZE_i$, the number of rows in $[h_F]_i$, ≤ 49 (must agree with order of $[M]_i$).

Field 2 contains NMODES, number of columns in $[h_F]_i$ (NFREE + $NCANT_i$). $NCANT_i$ is the number of cantilever modes to be input for surface i.

Field 3 contains NFORM; NFORM = 1 if the frequencies and the $[h_F]_i$ matrix elements are to be input using FORTRAN (FORMAT 6E12.8); NFORM = 0 if these data are input using column binary format.

b. Frequencies (ω_F series $_i$) (Format in above control card $_i$): If only cantilever modes are used for the system, an ω_F series is input following each control card $_i$. The ω_F's are listed in the order in which their respective modal columns will appear in $[h_F]_i$. (Number of ω_F's = NMODES, above card.)

If NFREE > 0, i = 1 for ω_F series$_i$, that is, only one ω_F series will be input for the system and this series must follow control card $_1$ (control card for the first surface). The number (≤ 40) of ω_F's in

the series must equal NFREE + NCANT used in Item 2. The ω_F's are listed in the order in which their respective modal columns appear in the unpartitioned $[h_F]$ matrix.

The ω_F series are input using the format specified in control card $_i$; NFORM = 1 (FORMAT 6E12.8) permits entering six ω_F's per card; use only the number of cards necessary to complete the series; NFORM = 0 (column binary format) should be used only if the ω_F's and modes are available as punched-card output from a suitable vibration program. When this format is used, the first ω_F starts in Origin 1; the number of ω_F's per card $\leq$ 22 and the ω_F card(s) $\leq$ 2 are followed by a TRA card.*

c. $[h_F]_i$ matrix elements (Format given in control card $_i$): When only cantilever modes are used for the system, the modal column order used in any one $[h_F]_i$ is independent of the order used for any other $[h_F]_i$, but if any free-free modes are used (NFREE > 0, Item 2), each free-free mode is partitioned for NSUR surfaces and the appropriate part of the mode must have the same column location in each $[h_F]_i$ matrix. The free-free modes must appear as the first NFREE columns in each $[h_F]_i$ matrix, the remaining columns (if any) are the cantilever modes peculiar to surface i. Use the format given in control card $_i$ and input $[h_F]_i$ by columns. If NFORM = 1, begin each column on a new card and use the number of cards ($\leq$9) necessary to complete the column. If NFORM = 0, each column begins in Origin 1 of a new card, the number of elements per card $\leq$ 22, and the card(s) $\leq$ 3 for each column are followed by a TRA card.

NOTE: Punched cards output from the IBM program based on Ref. 1 may be used for the above Parts a, b, and c with no alterations if the output data

---

*The TRA card has a 7 and a 9 punch in Column 1, Columns 2 through 72 are blank, and Columns 73 through 80 will contain the characters used for identification and sequencing in the punched-card output deck.

(cantilever or free-free modes and frequencies) are suitable for any component or the complete system considered in the modal analysis.

13. Weighting matrix, $[W]_i$, i = 1, NSUR: (No provisions have been made for entering a [W] matrix for the rigid component; any adjustments to $[C_{ho}]$ must be made before it is input.) The presence or absence of a [W] matrix for surface i is determined by ISW_i, Item 4; if $ISW_i = 1$, a [W] matrix must be input for surface i. When the matrix is input, it consists of two parts if control points for external stores have been reserved ($ISXT_i > 0$, Item 4). The input for $[W]_i$ matrix is as follows, with this exception: omit Parts a and b when $ISXT_i = 0$.

a. Control card $_i$ for external stores $[W]_i$ partition (FORMAT 18I4):

<u>Field 1</u> contains $NXST_i$; $NXST_i = 0$ if no $[W]_i$ matrix is input for the external stores area (the program will use a unit matrix, [I]); $NXST_i$ = number of control points reserved (= $ISXT_i$) if the partition is input.

<u>Field 2</u> is a dummy field and may be left blank.

<u>Field 3</u> contains NFORM; NFORM = 1 if the $[W]_i$ matrix elements will be input using FORTRAN (FORMAT 6E12.8); NFORM = 0 if the matrix elements will be input using column binary format.

<u>Field 4</u> contains NROW; NROW = 0 if the $[W]_i$ matrix elements will be input by column; NROW = 1 if the matrix elements will be input by row.

b. Elements in external stores $[W]_i$ partition (Format given in control card i): Omit this input if $NXST_i = 0$ in control card $_i$. If $NXST_i > 0$, input the matrix elements using the format combination given in control card $_i$; that is, if NFORM = 1 and NROW = 0, tabulate the elements

by column and begin each column on a new card, and if NFORM = 1 and NROW = 1 tabulate the elements by row with each row starting on a new card. When the column binary format is used (NFORM = 0 and NROW = 0) Column 1 starts in card Origin 1, Column 2 in Location (1 + $IXST_i$), Column 3 in Location (1 + 2 × $ISXT_i$), etc., and a TRA card must follow the input for this $[W]_i$ partition.

c. Control card $_i$ for component $[W]_i$ (FORMAT 18I4):

Field 1 contains $NSIZE_i$, number of control points used on surface i. (Do not include control points for external stores.)

Field 2 contains NPART, the number of partitions in the component $[W]_i$. NPART = 1 for a full (unpartitioned) matrix. (The $[W]_i$ matrix is often sparse, sometimes diagonal, and may be of rather large order, ≤ 49; for this reason, provision is made for partitioning the matrix and entering only the nonzero partitions.)

Field 3 contains NFORM; NFORM is defined above in Part a.

Field 4 contains NROW; NROW is also defined above in Part a.

The input below is repeated for NPART partitions (j = 1, NPART).*

d. Partition j size (FORMAT 18I4):

Field 1 contains the order of partition j (if NPART = 1, the order = $NSIZE_i$ used in above control card $_i$).

e. Elements in partition j (Format given in the above control card $_i$): The elements in each partition must be input using the format

*Part d is input only if NFORM = 1 (FORTRAN FORMAT) in the above control card; if NFORM = 0 (column binary format) the unpartitioned matrix is punched by columns and input as Part e.

combination given in Fields 3 and 4 of control card $_i$. See the explanation given in Part b (above) pertaining to the use of the format combinations.

14. Aerodynamic data: The aerodynamic input consists of NAERO (Item 2) sets of AICs. One set of AICs consists of the AICs for each surface which have the same reference $1/k_r$ (see Item 5). If NDENS = 0 (Item 2), a density series will precede each set of AICs. Input the density series (only if Item 7 was omitted) and the AIC matrices for each surface with the following input order. Repeat the order for each $(1/k)_j$, j = 1, NAERO.

a. Density series for $(1/k)_j$: (Omit this input if NDENS > 0.)

(a. 1) Control card $_j$ (FORMAT 18I4):

Field 1 contains NRHO, number of densities to be entered for $(1/k)_j$, ≤ 20.

(a. 2) Densities (FORMAT 6E12.8): Use the number of cards (≤ 4) necessary to complete the series.

b. Rigid component $[C_{ho}]_j$ matrix: (Omit this input if NFUS = 0, Item 2.)

(b. 1) Reference 1/k for $[C_{ho}]_j$ (FORMAT 6E12.8):

Field 1 in this card normally contains the 1/k used in computing $[C_{ho}]_j$. The 1/k card is used only to help identify $[C_{ho}]_j$ in the input deck and in the printed output; it may be a blank card, but the AICs in $[C_{ho}]_j$ must be computed for a 1/k which properly relates them to the j'th $1/k_r$. (See Items 5 and 6.)

(b. 2) Control card $_j$ for $[C_{ho}]_j$ (FORMAT 18I4):

Field 1 contains NSIZE, number of control points on the rigid component (must agree with NSIZE used for $[h_{Ro}]$).

Field 2 contains NPART, the number of partitions in $[C_{h_o}]_j$; NPART = 1 is used for an unpartitioned matrix.

Field 3 contains NFORM as defined in Item 13.a.

Field 4 contains NROW also defined in Item 13.a.

The input listed below must be repeated for NPART partitions.*

(b. 3) Partition k size, k = 1, NPART (FORMAT 18I4):

Field 1 in this card contains N, the order of partition k. If NPART = 1, N = NSIZE used in control card $_j$.

(b. 4) Elements in partition k (Format given in control card $_j$): All elements in the AIC matrices are complex numbers; but each partition is input as though it is a real matrix, size N × 2N. The imaginary parts of the elements form the even-numbered columns. Input each partition in $[C_{ho}]_j$ with the format combination specified in Fields 3 and 4 of the above control card. The format combinations are explained in Item 13. b.

c. $[C_h]_{ij}$ matrix for flexible surface i: The input order for $[C_h]_{ij}$ will differ from the order given for $[C_{ho}]_j$ only if external stores control points are reserved on surface i ($ISXT_i > 0$, Item 4). The input given below must be repeated for i = 1; NSUR and Parts c. 1 and c. 2 are omitted when no external stores are present on surface i.

***Part b. 3 is input only if NFORM = 1 (FORTRAN FORMAT) in the above control card; if NFORM = 0 (column binary format) the unpartitioned AIC matrix is punched by columns and input as Part b. 4.**

(c. 1) Control card $_{ij}$ for external stores $[C_h]_{ij}$ partition (FORMAT 18I4):

Field 1 contains $NXST_i$ = 0 if no $[C_h]_{ij}$ partition will be input. (The program sets the partition to zero.) $NXST_i$ = number of control points reserved for external stores if the external stores $[C_h]_{ij}$ is to be input.

Field 2 is a dummy field and may be left blank.

Field 3 contains NFORM as defined in Item 13. a.

Field 4 contains NROW as defined in Item 13. a.

(c. 2) Elements in external stores $[C_h]_{ij}$ partition (Format in above control card $_{ij}$): The elements in the partition are complex numbers and will be input in the same manner as the elements in the $[C_{ho}]_j$ partitions (Part b. 4, above), with the exception of format which is given in the above control card.

(c. 3) Reference 1/k for surface i (FORMAT 6E12. 8): (See Part b. 1, above.)

(c. 4) Control card $_{ij}$ for surface $[C_h]_{ij}$ partition (FORMAT 18I4): (See Part b. 2, above.)

The input below must be repeated for NPART partitions.*

(c. 5) Partition k size, k = 1, NPART (FORMAT 18I4): (See Part b. 3, above).

(c. 6) Elements in partition k: (Format in above control card, Part c. 4).

---

*See footnote page 37.

Note: Punched cards output from IBM programs based on the sources listed below may be used (with no alterations) for Item 14, Parts b.1, b.2, b.3, and b.4 or Parts c.3, c.4, c.5, and c.6 when the theory is appropriate to the problem. Punched cards output from the program based on slender-body theory may be used for Item 14, Parts c.1 and c.2 if the first card ($1/k_r$) is removed from the deck. The list shown is complete only at this writing; additional compatible sources are currently under development.

W. P. Rodden, E. F. Farkas, H. Malcom, and A. M. Kliszewski. "Aerodynamic Influence Coefficients from Incompressible Strip Theory: Analytical Development and Computational Procedure." Aerospace Corporation Report No. TDR-169(3230-11)TN-5, 3 September 1962.

W. P. Rodden, E. F. Farkas, H. Malcom, and A. M. Kliszewski. "Aerodynamic Influence Coefficients from Supersonic Strip Theory: Analytical Development and Computational Procedure." Aerospace Corporation Report No. TDR-169(3230-11)TN-1, 1 August 1962.

W. P. Rodden, E. F. Farkas, H. Malcom, and A. M. Kliszewski. "Aerodynamic Influence Coefficients from Piston Theory: Analytical Development and Computational Procedure." Aerospace Corporation Report No. TDR-169(3230-11)TN-2, 15 August 1962.

W. P. Rodden, E. F. Farkas, and G. Y. Takata. "Aerodynamic Influence Coefficients from Slender-Body Theory: Analytical Development and Computational Procedure." Aerospace Corporation Report No. TDR-169(3230-11)TN-6, 31 October 1962.

C. Example Keypunch Forms

Example keypunch forms are given on the following pages. Columns 73 through 80 are reserved for data deck identification and sequencing. Only the cards with sequencing are used in the two data decks set up for the example problems; the lines with Columns 73 through 80 left blank are for description of input data.

Title Card

```
JET TRANSPØRT EXAMPLE PRØBLEM — CANTILEVER INPUT MØDES                  HM140001
```

General Control Card (FORMAT 18I4)

```
   1   1   2   1   1   4   0   4   1   0   0                            HM140002
```

$ISXT_i$ and ISW_i, i = 1, NSUR (FORMAT 18I4)

```
   0   0                                                                HM140003
```

b_r and $(1/k_r)_j$ Series (FORMAT 6E12.8)

```
5.468       +16.67                                                      HM140004
```

b_{r_i} and s_i, i = 0, 1, NSUR (FORMAT 6E12.8)

```
5 468        18.9585    *5.468      37.917                              HM140005
```

Density Series (FORMAT 6E12.8)

```
+2378    -02                                                            HM140006
```

$[\Delta m]$ (FORMAT 6E12.8)

```
17400    +05 137025  +07                                                HM140007
137025   +07 4457907 2+10                                               HM140008
```

$NSIZE_i$ for $[M]_i$ (FORMAT 18I4)

```
  10                                                                    HM140009
```

LOW_j and $LHIGH_j$ for Each Column in $[M]_i$, j = 1, $NSIZE_i$ (FORMAT 18I4)

Columns 1–72	Columns 73–80
1 2 1 2 3 4 3 4 5 6 5 6 7 8 7 8 9 10	HM140010
9 10	HM140011
Elements in $[M]_i$ (FORMAT 6E12.8)	
5383.6 -134.9	HM140012
-134.9 925.2	HM140013
20732.0 -11005.0	HM140014
-11005.0 11478.0	HM140015
3113.9 139.7	HM140016
139.7 806.6	HM140017
2638.8 -21.0	HM140018
-21.0 803.3	HM140019
487.5 7.3	HM140020
7.3 177.9	HM140021
(Repeat the Above $[M]_i$ Input for i = 1, NSUR)	
Size (Number of Rows) for $[h_{Ro}]$ (FORMAT 18I4)	
5	HM140022
Elements in $[h_{Ro}]$ (FORMAT 6E12.8)	
1.0 1.0 1.0 1.0 1.0	HM140023

Cols 1–12	Cols 13–24	Cols 25–36	Cols 37–48	Cols 49–60	Cols 61–72	Cols 73–80
-373.30	-248.30	-123.30	+1.70	+126.70		HM140024

NSIZE$_i$ for $\left[h_R\right]_i$ (FORMAT 18I4)

Cols 1–12	Cols 13–24	Cols 25–36	Cols 37–48	Cols 49–60	Cols 61–72	Cols 73–80
10						HM140025

Elements in $\left[h_R\right]_i$ (FORMAT 6E12. 8)

Cols 1–12	Cols 13–24	Cols 25–36	Cols 37–48	Cols 49–60	Cols 61–72	Cols 73–80
1.0	1.0	1.0	1.0	1.0	1.0	HM140026
1.0	1.0	1.0	1.0			HM140027
20.25	-81.0	17.85	-71.40	15.80	-63.20	HM140028
13.30	-53.20	11.05	-44.20			HM140029

(Repeat the Above $\left[h_R\right]_i$ Input for i = 1, NSUR)

Control Card$_i$ for $\left[h_F\right]_i$ (FORMAT 18I4)

Cols 1–4	Cols 5–8	Cols 9–12	Cols 13–72	Cols 73–80
10	4	0		HM140030

Frequency (ω_F) Series for Surface i (FORMAT Given in Control Card)

Elements in $\left[h_F\right]_i$ Matrix (FORMAT Given in Control Card)

(The above control card indicates column binary format was used to input the ω's and the $\left[h_F\right]_i$ matrix; since these data are shown in the printed output (page 50), they will not be tabulated. To tabulate the data, use FORTRAN FORMAT 6E12. 8.)

1/k for Rigid Component $\left[C_{ho}\right]_j$ (FORMAT 6E12.8)

```
16.67                                                                   HM140041
```

Control Card $_j$ for $\left[C_{ho}\right]_j$ (FORMAT 18I4)

```
   5   1   0                                                            HM140042
```

(The above card indicates an unpartitioned 5 × 5 matrix to be input using column binary format.

The matrix can be seen in the printed output, page 51.)

1/k for Surface $\left[C_h\right]_{ij}$ (FORMAT 6E12.8)

```
16.67                                                                   HM140054
```

Control Card for $\left[C_h\right]_{ij}$ (FORMAT 18I4)

```
  10   5   0                                                            HM140055
```

(The above control card indicates a 10 × 10 matrix having five partitions and input using

column binary format. The matrix is shown in the printed output, pages 52-53.)

In the Above Data i = 1, NSUR and j = 1, NAERO (Number of $1/k_r$'s)

Title Card

```
JET TRANSPØRT EXAMPLE PRØBLEM - FREE-FREE INPUT MØDES                  HM141001
```

General Control Card (FORMAT 18I4)

```
   1   1   2   1   1   4   4   0   1   0   0                           HM141002
```

All other input for this problem appears in the preceding problem with the exception of the ω_y's and the $\left[h_y\right]_i$ matrix, and these can be seen in the printed output, page 63.

SECTION IV

PROGRAM OUTPUT

A. Printed Output

1. All input data.

2. The generalized mass matrices ($[m_{FF}]_i$ or $[m'_{FF}]$) corresponding to the input free vibration modal matrices ($[h_F]_i$ or $[h'_F]$).

3. The dynamic matrix if NPUNCH = -1 is used in the general control card.

4. For the vibration analysis or for each $1/k_r$ in the flutter analysis:

a. The eigenvalue for each output mode followed by the number of single- and double-precision iterations and the number of Aitken accelerations.

b. The eigenvectors for each mode followed by the check eigenvalues and eigenvectors.

c. The frequencies in cycles per second and, in a flutter analysis, the structural damping coefficient and the velocity (knots) associated with each frequency.

d. The final flutter or vibration modes for each component normalized on the largest element in each mode.

e. If NPUNCH = ±1, the sequencing numbers used in identifying the punched cards containing the frequencies and the final modes will appear below the respective printed data.

5. In a vibration analysis, the generalized mass corresponding to each output (free vibration) mode will follow the above printed output [see Eqs. (40) and (62)].

6. A number of different statements may be printed indicating machine- or program-detected errors in input data such as wrong format or incompatibility in the number of rows in $[h_R]_i$, $[M]_i$, and $[h_F]_i$.

7. If the program or the machine fails in the iteration portion of the program, a note will be printed stating the type or cause of failure. Some of the various program stops and machine failures are explained in Ref. 1, Appendix A.

8. If convergence is not obtained in the allowable number of iterations, a note will be printed and the program will continue. (In this case, the eigenvalues and eigenvectors should be compared with the check eigenvalues and eigenvectors to determine if the convergence is sufficiently accurate.)

9. The printed output for the two example problems is shown on the following pages.

JET TRANSPORT EXAMPLE PROBLEM-CANTILEVER INPUT MOOES HM140001

FLUTTER ANALYSIS BY A MODAL METHOO USING AEROOYNAMIC INFLUENCE COEFFICIENTS

NSUR = 1 NAERO = 1 NRIGIO = 2 NFUS = 1 NOENS = 1 MOOES OUT = 4 NFREE = 0 NCANT = 4

CANTILEVER INPUT MOOES

B RIGIO COMPONENT = 0.54680000E 01 S RIGIO COMPONENT = 0.18958499E 02

B (REF) = 0.54680000E 01

SURFACE	B	S	EXTERNAL STORES SIZE
1	0.54680000E 01	0.37916999E 02	0

RIGID COMPONENT MASS MATRIX

	COLUMN 1	COLUMN 2	COLUMN
1	0.17399999E 05	0.13702500E 07	
2	0.13702500E 07	0.44579072E 10	

MASS MATRIX

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.53835999E 04	-0.13490000E 03	0.	0.	0.	0.
2	-0.13490000E 03	0.92520000E 03	0.	0.	0.	0.
3	0.	0.	0.20732000E 05	-0.11004999E 05	0.	0.
4	0.	0.	-0.11004999E 05	0.11477999E 05	0.	0.
5	0.	0.	0.	0.	0.31139000E 04	0.13970000E 03
6	0.	0.	0.	0.	0.13970000E 03	0.80660000E 03
7	0.	0.	0.	0.	0.	0.
8	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.	0.	0.	0.	
2	0.	0.	0.	0.	
3	0.	0.	0.	0.	
4	0.	0.	0.	0.	
5	0.	0.	0.	0.	
6	0.	0.	0.	0.	
7	0.26387999E 04	-0.20999999E 02	0.	0.	
8	-0.20999999E 02	0.80329999E 03	0.	0.	
9	0.	0.	0.48750000E 03	0.72999999E 01	
10	0.	0.	0.72999999E 01	0.17789999E 03	

RIGIO COMPONENT MOOES, 5 CONTROL POINTS.

	COLUMN 1	COLUMN 2	COLUMN
1	0.C9999999E 01	-0.37329999E 03	
2	0.C9999999E 01	-0.24829999E 03	
3	0.09999999E 01	-0.12329999E 03	
4	0.09999999E 01	0.16999999E 01	
5	0.C9999999E 01	0.12670000E 03	

RIGID BODY MODAL MATRIX

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN
1	0.09999999E 01	0.20249999E 02	
2	0.09999999E 01	-0.80999999E 02	
3	0.09999999E 01	0.17850000E 02	
4	0.09999999E 01	-0.71399999E 02	
5	0.09999999E 01	0.15799999E 02	
6	0.09999999E 01	-0.63199999E 02	
7	0.09999999E 01	0.13299999E 02	
8	0.09999999E 01	-0.53200000E 02	
9	0.09999999E 01	0.11049999E 02	
10	0.09999999E 01	-0.44199999E 02	

FLEXIBLE MODAL MATRIX
SURFACE 1, 10 CONTROL POINTS

4 FREQUIENCIES

0.20368025E 01
0.35525894E 01
0.72804597E 01
0.11698543E 02

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN
1	0.45184767E-01	-0.16355576E-00	-0.21968562E-00	0.29141474E-00	
2	0.42502320E-01	0.54498983E 00	-0.24388658E-00	0.27963846E-00	
3	0.14825594E-00	-0.27963124E-00	-0.30546310E-00	0.30680391E-00	
4	0.14372702E-00	0.09999999E 01	-0.36288369E-00	0.35360781E-00	
5	0.37941088E-00	-0.22673982E-00	-0.37182412E-00	-0.55946033E 00	
6	0.37673284E-00	0.94656356E 00	-0.42861269E-00	-0.49016142E-00	
7	0.69299459E 00	-0.13888866E-00	0.20382352E-00	-0.14640462E-00	
8	0.69265460E 00	0.88787997E 00	0.16335642E-00	-0.20284958E-01	
9	0.99896891E 00	-0.39737375E-01	0.09999999E 01	0.83772944E 00	
10	0.09999999E 01	0.82236990E 00	0.98191517E 00	0.09999999E 01	

M FF MATRIX,SURFACE 1

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN
1	0.31500891E 04	0.15182495E-02	0.91362000E-03	0.90789795E-03	
2	0.14839172E-02	0.21329428E 05	-0.84495544E-03	-0.67901611E-03	
3	0.91552734E-03	-0.89263916E-03	0.27328113E 04	-0.32424927E-03	
4	0.90980530E-03	-0.59700012E-03	-0.29563904E-03	0.33404605E 04	

WEIGHTING MATRIX
SURFACE 1, NO WEIGHTING MATRIX

RIGID COMPONENT AERO MATRIX, 5 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.19883168E 01	-0.71028437E 01	-0.31782072E 01	0.99733821E 01	0.17128096E 01	-0.24681993E 01
2	0.20347044E 01	-0.30576088E 01	-0.89710928E 01	0.67589525E 00	0.25888789E 01	0.32829072E 01
3	0.	0.	0.25888789E 01	-0.32829072E 01	-0.89710928E 01	-0.67589525E 00
4	0.	0.	0.	0.	0.18608618E 01	-0.23997266E 01
5	0.	0.	0.	0.	0.19459838E-00	0.73566824E 00

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.	0.	0.	0.	
2	0.	0.	0.	0.	
3	0.20347044E 01	0.30576088E 01	0.	0.	
4	-0.59000403E 01	-0.46448669E-00	0.10709520E 01	0.17538263E 01	
5	0.45456574E-00	-0.31600294E 01	-0.19732837E 01	0.81508698E 00	

AERODYNAMIC MATRIX 1./K R = 0.16670000E 02

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.30099700E 03	-0.60092673E 02	-0.30579615E 03	0.32278208E 02	0.	0.
2	0.19111816E-00	0.16514119E 02	0.57335448E 00	-0.16514119E 02	0.	0.
3	0.	0.	0.	0.	0.29334980E 03	-0.55152775E 02
4	0.	0.	0.	0.	0.14211353E-00	0.13930788E 02
5	0.	0.	0.	0.	0.	0.
6	0.	0.	0.	0.	0.	0.
7	0.	0.	0.	0.	0.	0.
8	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN 11	COLUMN 12
1	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.
3	-0.29728241E 03	0.31246942E 02	0.	0.	0.	0.
4	0.42634058E-00	-0.13930788E 02	0.	0.	0.	0.
5	0.	0.	0.30483998E 03	-0.53868061E 02	-0.30827594E 03	0.31871554E 02
6	0.	0.	0.11384775E-00	0.12607991E 02	0.34154329E-00	-0.12607991E 02
7	0.	0.	0.	0.	0.	0.
8	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.

	COLUMN 13	COLUMN 14	COLUMN 15	COLUMN 16	COLUMN 17	COLUMN 18
1	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.
3	0.	0.	0.	0.	0.	0.
4	0.	0.	0.	0.	0.	0.
5	0.	0.	0.	0.	0.	0.
6	0.	0.	0.	0.	0.	0.
7	0.32483251E 03	-0.52278329E 02	-0.32767460E 03	0.32541166E 02	0.	0.
8	0.84216231E-01	0.11079562E 02	0.25264870E-00	-0.11079562E 02	0.	0.
9	0.	0.	0.	0.	0.30321474E 03	-0.43827882E 02
10	0.	0.	0.	0.	0.53236897E-01	0.84300261E 01

	COLUMN 19	COLUMN 20	COLUMN
1	0.	0.	
2	0.	0.	
3	0.	0.	
4	0.	0.	
5	0.	0.	
6	0.	0.	
7	0.	0.	
8	0.	0.	
9	-0.30521601E 03	0.28517172E 02	
10	0.15971068E-00	-0.84300261E 01	

OUTPUT DATA

FLUTTER ANALYSIS 8Y A MOOAL METHOO USING AERODYNAMIC INFLUENCE COEFFICIENTS

DENSITY = 0.23779999E-02 REOUCED VELOCITY = 0.16670000E 02

2 RIGIO BODY OEGREES OF FREEOOM, 4 INPUT MOOES (TOTAL).

MODE	EIGENVALUE		ITERATIONS S.P.	D.P.	AITKENS S.P.	D.P.
1	0.19004809E-00	-0.13066806E-00	11	0	3	0
2	0.13305187E-00	-0.90475378E-03	8	0	2	0
3	0.50936355E-02	-0.12321895E-01	11	0	3	0
4	0.89039633E-02	-0.15425324E-02	2	0	0	0

EIGENVECTORS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.C9999999E 01	0.	0.09999999E 01	-0.	0.23126761E-00	0.55586841E 00
2	-0.52882797E-02	0.26563946E-02	0.75557064E-02	-0.41647033E-01	0.18934382E-01	0.35449203E-02
3	0.46232747E-01	-0.26897484E-01	0.20981800E-01	-0.40106017E-01	0.09999999E 01	0.
4	-0.15751113E-02	-0.54060456E-02	-0.10034602E-01	-0.32852156E-02	0.35542952E-00	-0.13060754E-00
	COLUMN 7	COLUMN 8	COLUMN			
1	-0.50736564E 00	-0.28387121E-00				
2	0.31773394E-01	0.13081321E-01				
3	0.C9999999E 01	0.				
4	-0.15375598E-00	-0.25941110E-00				

CHECK EIGENVALUES AND EIGENVECTORS

```
0.19004808E-00 -0.13066801E-00  0.13305186E-00 -0.90471179E-03  0.50936326E-02 -0.12321895E-01
0.89039642E-02 -0.15425376E-02
```

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.09999999E 01	0.	0.09999999E 01	0.	0.23126674E-00	0.55586720E 00
2	-0.52882672E-02	0.26563954E-02	0.75557120E-02	-0.41647052E-01	0.18934361E-01	0.35450576E-02
3	0.46232741E-01	-0.26897469E-01	0.20981800E-01	-0.40105984E-01	0.09999999E 01	0.
4	-0.15751122E-02	-0.54060404E-02	-0.10034598E-01	-0.32852045E-02	0.35542956E-00	-0.13060750E-00

	COLUMN 7	COLUMN 8	COLUMN
1	-0.50736279E 00	-0.28387356E-00	
2	0.31773206E-01	0.13081333E-01	
3	0.09999999E 01	0.	
4	-0.15375577E-00	-0.25941128E-00	

MODE	OMEGA (CPS)	DAMPING	VELOCITY (KNOTS)
1	0.22938669E 01	-0.68755261E 00	0.77786975E 03
2	0.27415078E 01	-0.68000076E-02	0.92966856E 03
3	0.14011545E 02	-0.24190768E 01	0.47514340E 04
4	0.10597619E 02	-0.17324109E-00	0.35937425E 04

FINAL FLUTTER MOOES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.19844364E-00	0.78853738E-01	-0.93420087E-01	-0.97846834E-01	-0.39213833E-01	-0.41824473E-01
2	-0.21468721E-00	0.83936904E-01	-0.90291424E-01	-0.13011442E-00	-0.47020171E-01	-0.40065185E-01
3	-0.79901066E-01	0.77532722E-01	0.15015688E-01	-0.68443904E-01	-0.66332947E-01	-0.26054742E-02
4	-0.10292312E-00	0.85141466E-01	0.24666292E-01	-0.12520704E-00	-0.76256268E-01	-0.42543685E-02
5	0.19292901E-00	0.77061356E-01	0.27690162E-00	-0.16246606E-01	-0.22148340E-00	0.17809848E-00
6	0.17350294E-00	0.83907188E-01	0.28711402E-00	-0.67938515E-01	-0.22769985E-00	0.17440537E-00
7	0.59565143E 00	0.43911391E-01	0.63678547E 00	0.18358172E-01	0.25628873E-00	0.14763121E-00
8	0.58159092E 00	0.49055874E-01	0.64766992E 00	-0.27171946E-01	0.26902538E-00	0.13596687E-00
9	0.C9999999E 01	-0.	0.98947141E 00	0.38904484E-01	0.96910990E 00	0.18204748E-01
10	0.99054588E 00	0.33664029E-02	0.09999999E 01	0.	0.09999999E 01	-0.

	COLUMN 7	COLUMN 8	COLUMN
1	-0.50261029E-01	-0.15983910E-00	
2	-0.55549506E-01	-0.13875223E-00	
3	-0.16105451E-00	-0.32830670E-00	
4	-0.18329110E-00	-0.33670126E-00	
5	-0.35805747E-00	-0.21469803E-00	
6	-0.38073238E-00	-0.23812169E-00	
7	0.17717902E-00	-0.77397544E-01	
8	0.17168011E-00	-0.11724409E-00	
9	0.98417973E 00	0.42305075E-01	
10	0.C9999999E 01	0.	

FOR SIX CANTILEVER INPUT MODES:REDUCED VELOCITY = 16.67

MODE	OMEGA (CPS)	DAMPING	VELOCITY (KNOTS)
1	0.20945049E 01	-0.68642659E 00	0.71026438E 03
2	0.27472688E 01	0.42585590E-02	0.93162217E 03
3	0.48110338E 01	0.26282316E-00	0.16314623E 04
4	0.11158799E 02	-0.69910804E 00	0.37840436E 04
5	0.12543309E 02	-0.21431879E-00	0.42535426E 04

FINAL FLUTTER MODES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.22536646E-00	0.92572200E-01	-0.10430135E-00	-0.95571166E-01	-0.10669166E-00	-0.66641671E-02
2	-0.24172064E-00	0.96812297E-01	-0.10552717E-00	-0.12630315E-00	-0.13305146E-00	-0.51599176E-02
3	-0.10655960E-00	0.89932177E-01	0.75712425E-02	-0.67028530E-01	0.59504019E-01	-0.36529737E-01
4	-0.12957601E-00	0.95967677E-01	0.97156047E-02	-0.12113068E-00	0.20609725E-01	-0.37150735E-01
5	0.16918588E-00	0.83653867E-01	0.27401933E-00	-0.16638520E-01	0.42422621E-00	-0.10293659E-00
6	0.14333673E-00	0.94240405E-01	0.28014200E-00	-0.67000752E-01	0.42554127E-00	-0.12906936E-00
7	0.57887563E 00	0.49874357E-01	0.63759924E 00	0.19681936E-01	0.74136435E 00	-0.51632738E-01
8	0.54986458E 00	0.66028306E-01	0.64893137E 00	-0.26138963E-01	0.79228093E 00	-0.10830468E-00
9	0.09999999E 01	-0.	0.98852029E 00	0.39756269E-01	0.95080607E 00	0.52753998E-01
10	0.97653205E 00	0.13983364E-01	0.09999999E 01	0.	0.09999999E 01	-0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	-0.48560822E-03	-0.97199512E-01	0.55720084E 00	0.71647885E-01	
2	-0.54546969E-02	-0.86510264E-01	0.52352165E 00	0.59014091E-01	
3	-0.91827672E-01	-0.14610815E-00	0.90787979E 00	0.16462570E-01	
4	-0.97049317E-01	-0.15178238E-00	0.09999999E 01	-0.	
5	-0.47105751E-00	-0.21387362E-01	-0.11198328E-00	-0.33678662E-00	
6	-0.48660678E-00	-0.30987718E-01	-0.66459754E-01	-0.34726656E-00	
7	0.10400940E-00	0.10713105E-00	-0.39143629E-00	0.14244039E-00	
8	0.98177937E-01	0.96218284E-01	-0.39663463E-00	0.16197237E-00	
9	0.98493147E 00	0.29076613E-01	-0.28143807E-00	0.76281424E 00	
10	0.09999999E 01	0.	-0.26482295E-00	0.79094338E 00	

FOR EIGHT CANTILEVER INPUT MODES: REDUCED VELOCITY = 16.67

MODE	OMEGA (CPS)	DAMPING	VELOCITY (KNOTS)
1	0.20905757E 01	-0.69425995E 00	0.70893196E 03
2	0.27376900E 01	-0.92221314E-02	0.92837390E 03
3	0.46615379E 01	0.26804059E-00	0.15807670E 04
4	0.79630522E 01	-0.79058085E 00	0.27003385E 04
5	0.92712380E 01	-0.22886848E-00	0.31439554E 04

FINAL FLUTTER MODES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.22449498E-00	0.91537370E-01	-0.10730086E-00	-0.10117456E-00	-0.11255102E-00	-0.12529183E-02
2	-0.24101111E-00	0.94906926E-01	-0.10924226E-00	-0.13553381E-00	-0.14070798E-00	-0.75254908E-03
3	-0.10629883E-00	0.88999227E-01	0.51459261E-02	-0.71518499E-01	0.48310315E-01	-0.37257236E-01
4	-0.12895528E-00	0.94875598E-01	0.64448295E-02	-0.12697624E-00	0.13448917E-01	-0.40117770E-01
5	0.16779131E-00	0.83901346E-01	0.27362677E-00	-0.16119959E-01	0.41086898E-00	-0.13212080E-00
6	0.14339356E-00	0.92503455E-01	0.27768318E-00	-0.70304834E-01	0.40224075E-00	-0.14940493E-00
7	0.57771345E 00	0.50169168E-01	0.63765857E 00	0.20322871E-01	0.73138702E 00	-0.74002080E-01
8	0.54871025E 00	0.66334946E-01	0.64810055E 00	-0.26099673E-01	0.77613351E 00	-0.12960518E-00
9	0.09999999E 01	-0.	0.98828339E 00	0.37737560E-01	0.94867688E 00	0.58469196E-01
10	0.97549948E 00	0.15691465E-01	0.09999999E 01	0.	0.09999999E 01	-0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.34877675E-01	-0.57361844E-01	-0.12694786E-00	-0.13646928E-00	
2	0.19144013E-01	-0.37656026E-01	-0.15318850E-00	-0.55112755E 00	
3	-0.80863755E-01	-0.82678822E-01	-0.31536303E-00	-0.15939123E-00	
4	-0.86025547E-01	-0.64309233E-01	-0.31923633E-00	-0.53949848E-01	
5	-0.56713580E 00	-0.37153230E-01	-0.46245500E-00	-0.28981878E-00	
6	-0.53558409E 00	-0.78767737E-01	-0.49513949E-00	-0.27714869E-00	
7	-0.25098632E-01	0.38552232E-01	0.15663479E-00	-0.49470676E-01	
8	-0.35251486E-01	0.10751472E-01	0.13932018E-00	-0.10538456E-00	
9	0.09999999E 01	0.	0.98929078E 00	0.36008405E-01	
10	0.98434835E 00	-0.75285418E-02	0.09999999E 01	-0.	

FOR TEN CANTILEVER INPUT MODES:REDUCED VELOCITY = 16.67

MODE	OMEGA (CPS)	DAMPING	VELOCITY (KNOTS)
1	0.20648006E 01	-0.69443911E 00	0.70019143E 03
2	0.27409278E 01	-0.12059832E-01	0.92947187E 03
3	0.44713013E 01	0.27819017E-00	0.15162562E 04
4	0.69985107E 01	-0.61700474E 00	0.23732543E 04
5	0.95550425E 01	-0.26846905E-00	0.32401959E 04

FINAL FLUTTER MODES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.22672141E-00	0.92801609E-01	-0.10980555E-00	-0.10249897E-00	-0.11565813E-00	0.10797655E-02
2	-0.24322546E-00	0.96231766E-01	-0.11252964E-00	-0.13723268E-00	-0.14201696E-00	0.41647484E-03
3	-0.10896499E-00	0.90128804E-01	0.36838599E-02	-0.72438644E-01	0.39346464E-01	-0.37219568E-01
4	-0.13140173E-00	0.95914537E-01	0.38688469E-02	-0.12866598E-00	0.62708063E-02	-0.41508544E-01
5	0.16390745E-00	0.84669594E-01	0.27385022E-00	-0.15872125E-01	0.38962284E-00	-0.14034666E-00
6	0.13893208E-00	0.93805630E-01	0.27711197E-00	-0.70519595E-01	0.38136988E-00	-0.15814506E-00
7	0.57352667E 00	0.51462835E-01	0.63861454E 00	0.20940823E-01	0.72148301E 00	-0.93108028E-01
8	0.54571546E 00	0.66242959E-01	0.64748624E 00	-0.26460116E-01	0.75146683E 00	-0.13747214E-00
9	0.09999999E 01	-0.	0.98693995E 00	0.36698003E-01	0.94384036E 00	0.67121437E-01
10	0.97106527E 00	0.19206535E-01	0.09999999E 01	0.	0.09999999E 01	-0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.55524993E-01	-0.28636505E-01	0.10481205E-00	-0.75341270E-01	
2	0.37311652E-01	-0.33436552E-02	0.67771085E 00	-0.13639605E-00	
3	-0.61298352E-01	-0.81925987E-01	0.32304791E-00	-0.65253254E-01	
4	-0.72266892E-01	-0.58644155E-01	0.20867915E-00	-0.67861656E-01	
5	-0.53633323E 00	-0.13744034E-00	0.09999999E 01	-0.	
6	-0.51012327E 00	-0.16303784E-00	0.93896377E 00	0.10471679E-00	
7	-0.12072749E-00	-0.69964735E-01	-0.37512479E-00	0.53744053E 00	
8	-0.82757482E-01	-0.11890169E-00	-0.21398329E-00	0.38646627E-00	
9	0.09999999E 01	0.	-0.69760381E 00	-0.62525453E 00	
10	0.94309023E 00	-0.89771795E-02	-0.77933718E 00	-0.50264002E 00	

```
JET TRANSPORT EXAMPLE PROBLEM USING FREE-FREE INPUT MOOES          HM140001
          FLUTTER ANALYSIS BY A MODAL METHOD USING AEROOYNAMIC INFLUENCE COEFFICIENTS


NSUR =  1  NAERO =     1  NRIGIO =  2  NFUS =  1  NOENS =     1  MOOES OUT =  4  NFREE =    4  NCANT =    0

              FREE-FREE INPUT MOOES,     0 CANTILEVER MOOES INCLUDED.

     B RIGID COMPONENT =      0.54680000E 01      S RIGID COMPONENT =     0.18958499E 02

                          B (REF) =        0.54680000E 01

            SURFACE                      B                    S          EXTERNAL STORES SIZE


               1          0.54680000E 01          0.37916999E 02           -0

                              RIGID COMPONENT MASS MATRIX
     COLUMN    1       COLUMN    2       COLUMN
1   0.17399999E 05   0.13702500E 07
2   0.13702500E 07   0.44579072E 10
```

MASS MATRIX

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.53835999E 04	-0.13490000E 03	0.	0.	0.	0.
2	-0.13490000E 03	0.92520000E 03	0.	0.	0.	0.
3	0.	0.	0.20732000E 05	-0.11004999E 05	0.	0.
4	0.	0.	-0.11004999E 05	0.11477999E 05	0.	0.
5	0.	0.	0.	0.	0.31139000E 04	0.13970000E 03
6	0.	0.	0.	0.	0.13970000E 03	0.80660000E 03
7	0.	0.	0.	0.	0.	0.
8	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.	0.	0.	0.	
2	0.	0.	0.	0.	
3	0.	0.	0.	0.	
4	0.	0.	0.	0.	
5	0.	0.	0.	0.	
6	0.	0.	0.	0.	
7	0.26387999E 04	-0.20999999E 02	0.	0.	
8	-0.20999999E 02	0.80329999E 03	0.	0.	
9	0.	0.	0.48750000E 03	0.72999999E 01	
10	0.	0.	0.72999999E 01	0.17789999E 03	

RIGID COMPONENT MODES, 5 CONTROL POINTS.

	COLUMN 1	COLUMN 2	COLUMN
1	0.C9999999E 01	-0.37329999E 03	
2	0.09999999E 01	-0.24829999E 03	
3	0.C9999999E 01	-0.12329999E 03	
4	0.C9999999E 01	0.16999999E 01	
5	0.C9999999E 01	0.12670000E 03	

RIGID BODY MODAL MATRIX
SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN
1	0.09999999E 01	0.20249999E 02	
2	0.C9999999E 01	-0.80999999E 02	
3	0.C9999999E 01	0.17850000E 02	
4	0.C9999999E 01	-0.71399999E 02	
5	0.09999999E 01	0.15799999E 02	
6	0.C9999999E 01	-0.63199999E 02	
7	0.C9999999E 01	0.13299999E 02	
8	0.C9999999E 01	-0.53200000E 02	
9	0.09999999E 01	0.11049999E 02	
10	0.C9999999E 01	-0.44199999E 02	

FLEXIBLE MODAL MATRIX
SURFACE 1, 10 CONTROL POINTS

4 FREQUIENCIES

0.24429615E 01
0.36031128E 01
0.85530452E 01
0.12707073E 02

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN
1	-0.12972128E-00	-0.91809043E-01	-0.35610154E-01	0.30597945E-00	
2	-0.13133238E-00	0.56810184E 00	-0.47615869E-01	0.28780279E-00	
3	-0.17536534E-01	-0.22162712E-00	-0.16294216E-00	0.44284272E-00	
4	-0.15998174E-01	0.09999999E 01	-0.18977508E-00	0.50378348E 00	
5	0.24780568E-00	-0.21537105E-00	-0.44938723E-00	-0.56146173E 00	
6	0.25089253E-00	0.90554103E 00	-0.47773996E-00	-0.44347260E-00	
7	0.62266540E 00	-0.20513079E-00	0.10694845E-00	-0.22509799E-00	
8	0.62794039E 00	0.77614732E 00	0.97215801E-01	-0.62188134E-02	
9	0.99374234E 00	-0.18689994E-00	0.98440876E 00	0.74102239E 00	
10	0.09999999E 01	0.63668663E 00	0.09999999E 01	0.09999999E 01	

M FF MATRIX, SURFACE

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN
1	0.28519310E 04	-0.19187927E-02	-0.44631958E-03	-0.54168701E-03	
2	-0.19111633E-02	0.19691394E 05	0.18501282E-02	0.18329620E-02	
3	-0.44631958E-03	0.18501282E-02	0.23241071E 04	0.85449219E-03	
4	-0.53405762E-03	0.18367767E-02	0.86212158E-03	0.54105940E 04	

WEIGHTING MATRIX

SURFACE 1,   NO WEIGHTING MATRIX

RIGIO COMPONENT AERO MATRIX,   5 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.19883168E 01	-0.71028437E 01	-0.31782072E 01	0.99733821E 01	0.17128096E 01	-0.24681993E 01
2	0.20347044E 01	-0.30576088E 01	-0.89710928E 01	0.67589525E 00	0.25888789E 01	0.32829072E 01
3	0.	0.	0.25888789E 01	-0.32829072E 01	-0.89710928E 01	-0.67589525E 00
4	0.	0.	0.	0.	0.18608618E 01	-0.23997266E 01
5	0.	0.	0.	0.	0.19459838E-00	0.73566824E 00

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.	0.	0.	0.	
2	0.	0.	0.	0.	
3	0.20347044E 01	0.30576088E 01	0.	0.	
4	-0.59000403E 01	-0.46448669E-00	0.10709520E 01	0.17538263E 01	
5	0.45456574E-00	-0.31600294E 01	-0.19732837E 01	0.81508698E 00	

AERODYNAMIC MATRIX 1./K R = 0.16670000E 02

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.30099700E 03	-0.60092673E 02	-0.30579615E 03	0.32278208E 02	0.	0.
2	0.19111816E-00	0.16514119E 02	0.57335448E 00	-0.16514119E 02	0.	0.
3	0.	0.	0.	0.	0.29334980E 03	-0.55152775E 02
4	0.	0.	0.	0.	0.14211353E-00	0.13930788E 02
5	0.	0.	0.	0.	0.	0.
6	0.	0.	0.	0.	0.	0.
7	0.	0.	0.	0.	0.	0.
8	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN 11	COLUMN 12
1	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.
3	-0.29728241E 03	0.31246942E 02	0.	0.	0.	0.
4	0.42634058E-00	-0.13930788E 02	0.	0.	0.	0.
5	0.	0.	0.30483998E 03	-0.53868061E 02	-0.30827594E 03	0.31871554E 02
6	0.	0.	0.11384775E-00	0.12607991E 02	0.34154329E-00	-0.12607991E 02
7	0.	0.	0.	0.	0.	0.
8	0.	0.	0.	0.	0.	0.
9	0.	0.	0.	0.	0.	0.
10	0.	0.	0.	0.	0.	0.

	COLUMN 13	COLUMN 14	COLUMN 15	COLUMN 16	COLUMN 17	COLUMN 18
1	0.	0.	0.	0.	0.	0.
2	0.	0.	0.	0.	0.	0.
3	0.	0.	0.	0.	0.	0.
4	0.	0.	0.	0.	0.	0.
5	0.	0.	0.	0.	0.	0.
6	0.	0.	0.	0.	0.	0.
7	0.32483251E 03	-0.52278329E 02	-0.32767460E 03	0.32541166E 02	0.	0.
8	0.84216231E-01	0.11079562E 02	0.25264870E-00	-0.11079562E 02	0.	0.
9	0.	0.	0.	0.	0.30321474E 03	-0.43827882E 02
10	0.	0.	0.	0.	0.53236897E-01	0.84300261E 01

	COLUMN 19	COLUMN 20	COLUMN
1	0.	0.	
2	0.	0.	
3	0.	0.	
4	0.	0.	
5	0.	0.	
6	0.	0.	
7	0.	0.	
8	0.	0.	
9	-0.30521601E 03	0.28517172E 02	
10	0.15971068E-00	-0.84300261E 01	

OUTPUT DATA

FLUTTER ANALYSIS BY A MODAL METHOD USING AERODYNAMIC INFLUENCE COEFFICIENTS

DENSITY = 0.23779999E-02 REDUCED VELOCITY = 0.16670000E 02

2 RIGID BODY DEGREES OF FREEDOM, 4 INPUT MODES (TOTAL).

MODE	EIGENVALUE		ITERATIONS S.P.	D.P.	AITKENS S.P.	D.P.
1	0.18916545E-00	-0.13115912E-00	11	0	3	0
2	0.13292938E-00	-0.86249603E-03	8	0	2	0
3	0.47592145E-02	-0.12187260E-01	14	0	4	0
4	0.98496465E-02	-0.14207529E-02	2	0	0	0

EIGENVECTORS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.09999999E 01	0.	0.09999999E 01	-0.	0.44428806E-00	0.50267780E 00
2	-0.17784069E-01	0.34797354E-02	-0.17137665E-02	-0.48118412E-01	0.96034528E-03	-0.22564518E-02
3	0.15017853E-01	-0.29673016E-01	-0.13965693E-01	-0.43682259E-01	0.09999999E 01	0.
4	0.17615897E-02	-0.10034880E-02	-0.19148218E-02	0.23242460E-02	0.14828939E-00	-0.69422592E-01
	COLUMN 7	COLUMN 8	COLUMN			
1	-0.12961736E-00	-0.28763704E-00				
2	0.18731757E-01	0.18930750E-01				
3	0.09999999E 01	0.				
4	-0.17362598E-00	-0.19799793E-00				

CHECK EIGENVALUES AND EIGENVECTORS

0.18916556E-00 -0.13115912E-00 0.13292948E-00 -0.86250604E-03 0.47592180E-02 -0.12187257E-01
0.98496455E-02 -0.14207555E-02

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	0.09999999E 01	0.	0.09999999E 01	0.	0.44428635E-00	0.50268287E 00
2	-0.17784079E-01	0.34797155E-02	-0.17138086E-02	-0.48118390E-01	0.96057817E-03	-0.22566880E-02
3	0.15017867E-01	-0.29672989E-01	-0.13965622E-01	-0.43682230E-01	0.09999999E 01	0.
4	0.17615926E-02	-0.10034867E-02	-0.19148137E-02	0.23242421E-02	0.14828938E-00	-0.69422607E-01

	COLUMN 7	COLUMN 8	COLUMN
1	-0.12961851E-00	-0.28764046E-00	
2	0.18731755E-01	0.18930972E-01	
3	0.09999999E 01	0.	
4	-0.17362595E-00	-0.19799802E-00	

MODE	OMEGA (CPS)	DAMPING	VELOCITY (KNOTS)
1	0.22992123E 01	-0.69335660E 00	0.77968240E 03
2	0.27427707E 01	-0.64883775E-02	0.93009681E 03
3	0.14495472E 02	-0.25607714E 01	0.49155374E 04
4	0.10076035E 02	-0.14424405E-00	0.34168689E 04

FINAL FLUTTER MODES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.19798382E-00	0.79322939E-01	-0.93759178E-01	-0.98772941E-01	-0.38185357E-01	-0.39426726E-01
2	-0.21410581E-00	0.84471434E-01	-0.90696646E-01	-0.13123662E-00	-0.46656333E-01	-0.37693791E-01
3	-0.79324013E-01	0.78083789E-01	0.15005647E-01	-0.68983589E-01	-0.64005999E-01	-0.28086398E-02
4	-0.10239267E-00	0.85738368E-01	0.24328708E-01	-0.12634239E-00	-0.74012243E-01	-0.39014272E-02
5	0.19340336E-00	0.77492066E-01	0.27696662E-00	-0.16448494E-01	-0.21635852E-00	0.16610579E-00
6	0.17411190E-00	0.84444531E-01	0.28701860E-00	-0.68477032E-01	-0.22248337E-00	0.16209281E-00
7	0.59585255E 00	0.44114260E-01	0.63675676E 00	0.18233940E-01	0.25981362E-00	0.13876332E-00
8	0.58214916E 00	0.49438009E-01	0.64768282E 00	-0.27309294E-01	0.27279337E-00	0.12600318E-00
9	0.09999999E 01	-0.	0.98938049E 00	0.38838806E-01	0.96899516E 00	0.18922722E-01
10	0.99091579E 00	0.35426569E-02	0.09999999E 01	0.	0.09999999E 01	-0.

COLUMN

	COLUMN 7	COLUMN 8
1	-0.42771401E-01	-0.12797226E-00
2	-0.50017554E-01	-0.11050270E-00
3	-0.15618632E-00	-0.30295999E-00
4	-0.17657918E-00	-0.30384143E-00
5	-0.37818480E-00	-0.29088473E-00
6	-0.40172405E-00	-0.31386279E-00
7	0.16239082E-00	-0.13932644E-00
8	0.15330823E-00	-0.18515402E-00
9	0.98711184E 00	0.45538359E-01
10	0.C9999999E 01	0.

FOR SIX FREE-FREE INPUT MODES:REDUCED VELOCITY = 16.67

MODE	OMEGA (CPS)	DAMPING	VELOCITY (KNOTS)
1	0.20944695E 01	-0.68661743E 00	0.71025240E 03
2	0.27474146E 01	0.47400708E-02	0.93167160E 03
3	0.48123113E 01	0.26238423E-00	0.16318955E 04
4	0.11123481E 02	-0.69183040E 00	0.37720668E 04
5	0.12642173E 02	-0.22247387E-00	0.42870682E 04

FINAL FLUTTER MODES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.22536416E-00	0.92623165E-01	-0.10428888E-00	-0.95617969E-01	-0.10694816E-00	-0.64947930E-02
2	-0.24158062E-00	0.96910723E-01	-0.10536446E-00	-0.12621076E-00	-0.13273688E-00	-0.52423403E-02
3	-0.10638769E-00	0.90016660E-01	0.77892566E-02	-0.66937186E-01	0.60163083E-01	-0.36864151E-01
4	-0.12939467E-00	0.96067116E-01	0.99430882E-02	-0.12104959E-00	0.21114814E-01	-0.37425315E-01
5	0.16926584E-00	0.83706391E-01	0.27414474E-00	-0.16616217E-01	0.42480908E-00	-0.10347813E-00
6	0.14343195E-00	0.94309451E-01	0.28028685E-00	-0.66988336E-01	0.42607864E-00	-0.12961110E-00
7	0.57890615E 00	0.49899049E-01	0.63765150E 00	0.19684771E-01	0.74169123E 00	-0.51988563E-01
8	0.54990779E 00	0.66068787E-01	0.64900336E 00	-0.26153403E-01	0.79264993E 00	-0.10870654E-00
9	0.09999999E 01	-0.	0.98849916E 00	0.39769159E-01	0.95074619E 00	0.52796511E-01
10	0.97654707E 00	0.13997710E-01	0.09999999E 01	0.	0.09999999E 01	-0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	-0.18877905E-02	-0.96578937E-01	0.55185816E 00	0.72051873E-01	
2	-0.74029310E-02	-0.87815752E-01	0.53127283E 00	0.59908347E-01	
3	-0.95516861E-01	-0.14733771E-00	0.91057707E 00	0.15560011E-01	
4	-0.10089871E-00	-0.15260053E-00	0.09999999E 01	-0.	
5	-0.47069521E-00	-0.22004030E-01	-0.13254518E-00	-0.34762463E-00	
6	-0.48633242E-00	-0.31481614E-01	-0.88586425E-01	-0.35831629E-00	
7	0.10496839E-00	0.10617295E-00	-0.38397461E-00	0.14641586E-00	
8	0.99158280E-01	0.95083151E-01	-0.38881926E-00	0.16508962E-00	
9	0.98502792E 00	0.29177163E-01	-0.23295638E-00	0.77594201E 00	
10	0.09999999E 01	0.	-0.21485166E-00	0.80241252E 00	

FOR EIGHT FREE-FREE INPUT MODES:REDUCED VELOCITY = 16.67

MOOE	OMEGA (CPS)	OAMPING	VELOCITY (KNOTS)
1	0.20907349E 01	-0.69434052E 00	0.70898596E 03
2	0.27376843E 01	-0.94513867E-02	0.92837197E 03
3	0.46601459E 01	0.26838358E-00	0.15802950E 04
4	0.79644649E 01	-0.79140406E 00	0.27008176E 04
5	0.92728179E 01	-0.22534730E-00	0.31444912E 04

FINAL FLUTTER MODES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.22462561E-00	0.91470455E-01	-0.10749614E-00	-0.10132902E-00	-0.11310212E-00	-0.10736525E-02
2	-0.24121621E-00	0.94815151E-01	-0.10954312E-00	-0.13576087E-00	-0.14155165E-00	-0.36489125E-03
3	-0.10628965E-00	0.88974198E-01	0.51262185E-02	-0.71538554E-01	0.48264105E-01	-0.37285466E-01
4	-0.12894306E-00	0.94845173E-01	0.64150822E-02	-0.12699864E-00	0.13509803E-01	-0.40200412E-01
5	0.16775496E-00	0.83875872E-01	0.27355848E-00	-0.16161098E-01	0.41046529E-00	-0.13178461E-00
6	0.14335689E-00	0.92471910E-01	0.27759990E-00	-0.70349848E-01	0.40187399E-00	-0.14909881E-00
7	0.57769161E 00	0.50157983E-01	0.63762022E 00	0.20299893E-01	0.73113547E 00	-0.73786326E-01
8	0.54868171E 00	0.66317197E-01	0.64803984E 00	-0.26127829E-01	0.77580259E 00	-0.12934569E-00
9	0.09999999E 01	-0.	0.98827996E 00	0.37722594E-01	0.94868443E 00	0.58454064E-01
10	0.97551249E 00	0.15691373E-01	0.09999999E 01	0.	0.09999999E 01	-0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.35467197E-01	-0.56447988E-01	-0.12322193E-00	-0.14081804E-00	
2	0.19926880E-01	-0.35919973E-01	-0.15028110E-00	-0.55685750E 00	
3	-0.80843733E-01	-0.82540129E-01	-0.31446418E-00	-0.16249044E-00	
4	-0.86095372E-01	-0.64346923E-01	-0.31816854E-00	-0.57204825E-01	
5	-0.56675266E 00	-0.36877990E-01	-0.46181442E-00	-0.28872070E-00	
6	-0.53520397E 00	-0.78556748E-01	-0.49441168E-00	-0.27616097E-00	
7	-0.24862114E-01	0.38486993E-01	0.15692800E-00	-0.47496054E-01	
8	-0.34927980E-01	0.10759114E-01	0.13985911E-00	-0.10349906E-00	
9	0.09999999E 01	0.	0.98930217E 00	0.35894447E-01	
10	0.98434630E 00	-0.75488728E-02	0.09999999E 01	-0.	

FOR TEN FREE-FREE INPUT MODES: REDUCED VELOCITY = 16.67

MOOE	OMEGA (CPS)	DAMPING	VELOCITY (KNOTS)
1	0.20647985E 01	-0.69443611E 00	0.70019069E 03
2	0.27409285E 01	-0.12062252E-01	0.92947210E 03
3	0.44713092E 01	0.27819197E-00	0.15162589E 04
4	0.69985186E 01	-0.61700870E 00	0.23732570E 04
5	0.95550377E 01	-0.26846989E-00	0.32401942E 04

FINAL FLUTTER MODES

SURFACE 1, 10 CONTROL POINTS

	COLUMN 1	COLUMN 2	COLUMN 3	COLUMN 4	COLUMN 5	COLUMN 6
1	-0.22672210E-00	0.92801184E-01	-0.10980548E-00	-0.10249859E-00	-0.11565738E-00	0.10802589E-02
2	-0.24322628E-00	0.96231290E-01	-0.11252954E-00	-0.13723224E-00	-0.14201599E-00	0.41718773E-03
3	-0.10896556E-00	0.90128426E-01	0.36838976E-02	-0.72438279E-01	0.39346816E-01	-0.37219467E-01
4	-0.13140251E-00	0.95914066E-01	0.38689399E-02	-0.12866554E-00	0.62714847E-02	-0.41508120E-01
5	0.16390713E-00	0.84669287E-01	0.27385023E-00	-0.15871787E-01	0.38962222E-00	-0.14034727E-00
6	0.13893154E-00	0.93805225E-01	0.27711201E-00	-0.70519144E-01	0.38136957E-00	-0.15814554E-00
7	0.57352652E 00	0.51462667E-01	0.63861449E 00	0.20940991E-01	0.72148240E 00	-0.93108623E-01
8	0.54571509E 00	0.66242675E-01	0.64748625E 00	-0.26459806E-01	0.75146637E 00	-0.13747277E-00
9	0.09999999E 01	-0.	0.98693993E 00	0.36697865E-01	0.94384047E 00	0.67121518E-01
10	0.97106511E 00	0.19206408E-01	0.09999999E 01	0.	0.09999999E 01	-0.

	COLUMN 7	COLUMN 8	COLUMN 9	COLUMN 10	COLUMN
1	0.55525421E-01	-0.28637560E-01	0.10480621E-00	-0.75338227E-01	
2	0.37312139E-01	-0.33451135E-02	0.67769676E 00	-0.13640756E-00	
3	-0.61299051E-01	-0.81927439E-01	0.32303939E-00	-0.65244763E-01	
4	-0.72267612E-01	-0.58646142E-01	0.20867298E-00	-0.67849804E-01	
5	-0.53633708E 00	-0.13744275E-00	0.09999999E 01	-0.	
6	-0.51012741E 00	-0.16304075E-00	0.93896465E 00	0.10472411E-00	
7	-0.12073137E-00	-0.69966570E-01	-0.37512115E-00	0.53746350E 00	
8	-0.82761717E-01	-0.11890361E-00	-0.21398164E-00	0.38648304E-00	
9	0.09999999E 01	0.	-0.69761330E 00	-0.62528355E 00	
10	0.94308997E 00	-0.89770386E-02	-0.77934656E 00	-0.50266519E 00	

B. Punched Output

1. A deck of punched cards is output from the vibration analysis portion of this program that is suitable as input to other programs requiring free-free vibration modes and frequencies. The flutter modes and frequencies also may be punched in cards if it is so desired. If a printout of the dynamic matrix is needed (NPUNCH = -1) but no punched-card output is wanted, the machine operator must be instructed to ignore the output punch tape.

2. All punched output is sequenced in order on Column 73 through 80 starting with HM140000. Each mode will have NSUR (number of surfaces) partitions, and the output data will be in the following order.

a. Control card for surface 1 (FORMAT 18I4):

Field 1 contains $NSIZE_1$, number of control points on component 1.

Field 2 contains NMODES, number of output modes punched.

b. Frequencies (column binary format): The frequencies are punched in the order in which they appear in the frequency column in the printed output. The column binary card(s) (only one card if NMODES $\leq$ 22) containing the frequencies are followed by a TRA card.

c. Modes for component 1 (column binary format): The component 1 partition of each vibration mode is punched as a column with a TRA card following the output for each column. If the final flutter modes are punched in cards, a TRA card will follow both the output for the real column and the output for the imaginary column forming each mode.

d. Control cards and modes for component i, i = 2, NSUR: Parts a and c (above) are repeated for NSUR surfaces when NSUR > 1.

SECTION V

PROCESSING INFORMATION

A. Operation

FORTRAN II MONITOR system (uses chain job capability).

B. Estimated Machine Time

Because of the many variables involved such as number of input modes, number of surfaces, number of iterations required for eigenvector convergence, etc., it is not possible to give a practical method of estimating machine time. Complete but independent decks may be stacked to save program read-in time. The two data decks for the example problems were stacked and the machine time was 1.84 minutes.

C. Machine Components Used

The following machine logical tape units were used; each is set equal to the symbolic tape unique to the program.

TAPE 2 (input tape) = NTAPE2

TAPE 3 (output print tape) = NTAPE3

TAPE 11 (chain tape) = NTAPE1

TAPE 12 (output punch tape) = NTAPE0

TAPES 13, 5, 6, 7, 8, and 14 (utility tapes) = NTAPES4, 5, 6, 7, 8, and 9, respectively.

NTAPE1 contains two chains. Chain 1 uses six utility tapes for storing matrices; the number of rows and number of columns precedes each matrix. The symbolic utility tapes and the matrices stored on each are listed below (i = 1, NSUR on all tapes except NTAPE 9).

NTAPE4: 1) $[M]_i$

If NRIGID > 0

2) $[h_R]$ for the system

If NRIGID > 0 and NFREE > 0

3) $[a_{RF}]$

NTAPE5: 1) $[h_F]_i$

NTAPE6: If NRIGID > 0 and NFUS > 0

1) $[h_{Ro}]$

If any $ISW_i > 0$

2) $[W]_i$

NTAPE7: 1) $[W]_i[C_h]_i$ or $[C_h]_i$

NTAPE8: If NRIGID > 0

1) $[\overline{M}]_i = \left([M]_i + 32.174\rho_j b_{ri}^2 s_i [W]_i [C_h]_i\right)$
j = 1, NDENS or 1, NRHO

2) $[\overline{M}]_i[h_F]_i$

3) $[h_F]_i^T[\overline{M}]_i$

NTAPE9: 1) $[\overline{m}_{FF}] = [h_F]_i^T[\overline{M}]_i[h_F]_i$, $i = 1$ for $[h_F]_i$
if any free-free modes are input;

If NRIGID > 0

2) $[\overline{m}_{RF}]_i = [h_R]_i^T[\overline{M}]_i[h_F]_i$

3) $[\overline{m}_{FR}]_i = [h_F]_i^T[\overline{M}]_i[h_R]_i$

If NRIGID > 0 and NFREE > 0

4) $[\Delta\overline{m}][a_{RF}]$

5) $[a_{RF}]^T[\Delta\overline{m}]$

6) $[a_{RF}]^T[\Delta\overline{m}][a_{RF}]$

If NRIGID > 0

7) $[\overline{m}_{RR}]^{-1} = \left([h_R]^T[\overline{M}][h_R] + [\Delta\overline{m}]\right)^{-1}$

8) $[h_R][\overline{m}_{RR}]^{-1}$

Chain 2 also uses NTAPE8 for storing computations used in the MITERS subroutine and for storing the normalized vibration or flutter modes for each surface.

SECTION VI

PROGRAM NOTES

A. Subroutine Used

In addition to the main program (page 91), the following subroutines were used.

1. RDLN, reads and prints the title card (page 127).

2. MMULTD, computes the product [C] of two matrices [A] and [B] (page 129).

3. MPRINT, prints matrix in matrix format (page 132).

4. MREAD, reads a matrix in either column binary or FORTRAN format (page 135).

5. BINRD, reads column binary cards from tape (page 140).

6. MNVRSX, computes the inverse of a complex matrix (page 150).

7. INVERS, computes the inverse of a real matrix (page 153).

8. BINPU, writes column binary cards on tape (page 158).

9. MPUNCH, punches column binary cards (page 168).

10. MITERS, real or complex matrix iteration (page 172).

11. NPNRMX, vector normalization (page 185).

12. SWEEPX, computes the true mode and sweeps it from the related matrix (page 190).

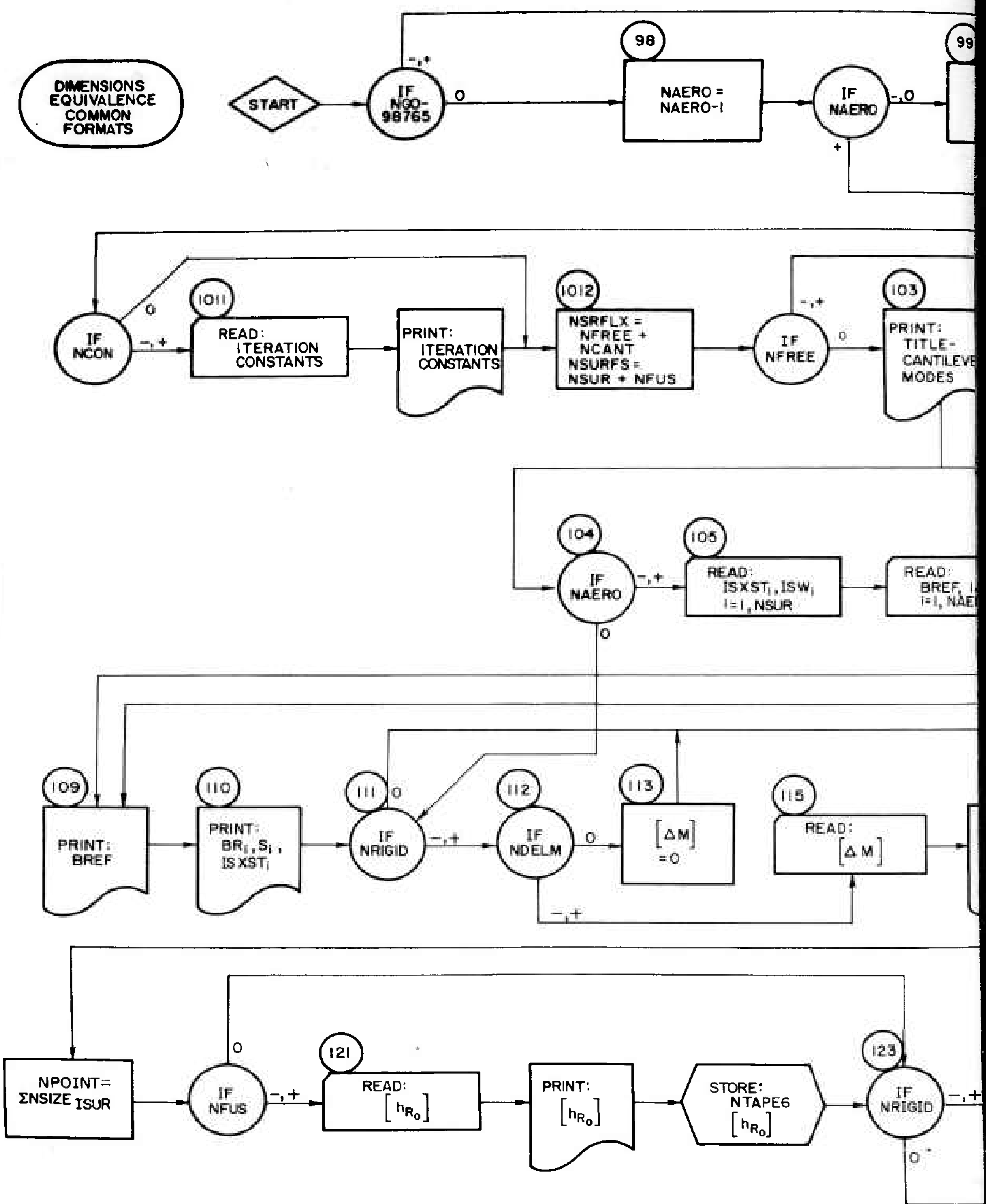
13. CLOSE, matrix iteration using double-precision arithmetic (page 196).

14. DPFORM, converts double-precision numbers to single-precision numbers and conversely (page 207).

15. DPMLTX, matrix multiplication using single- or double-precision numbers (page 211).

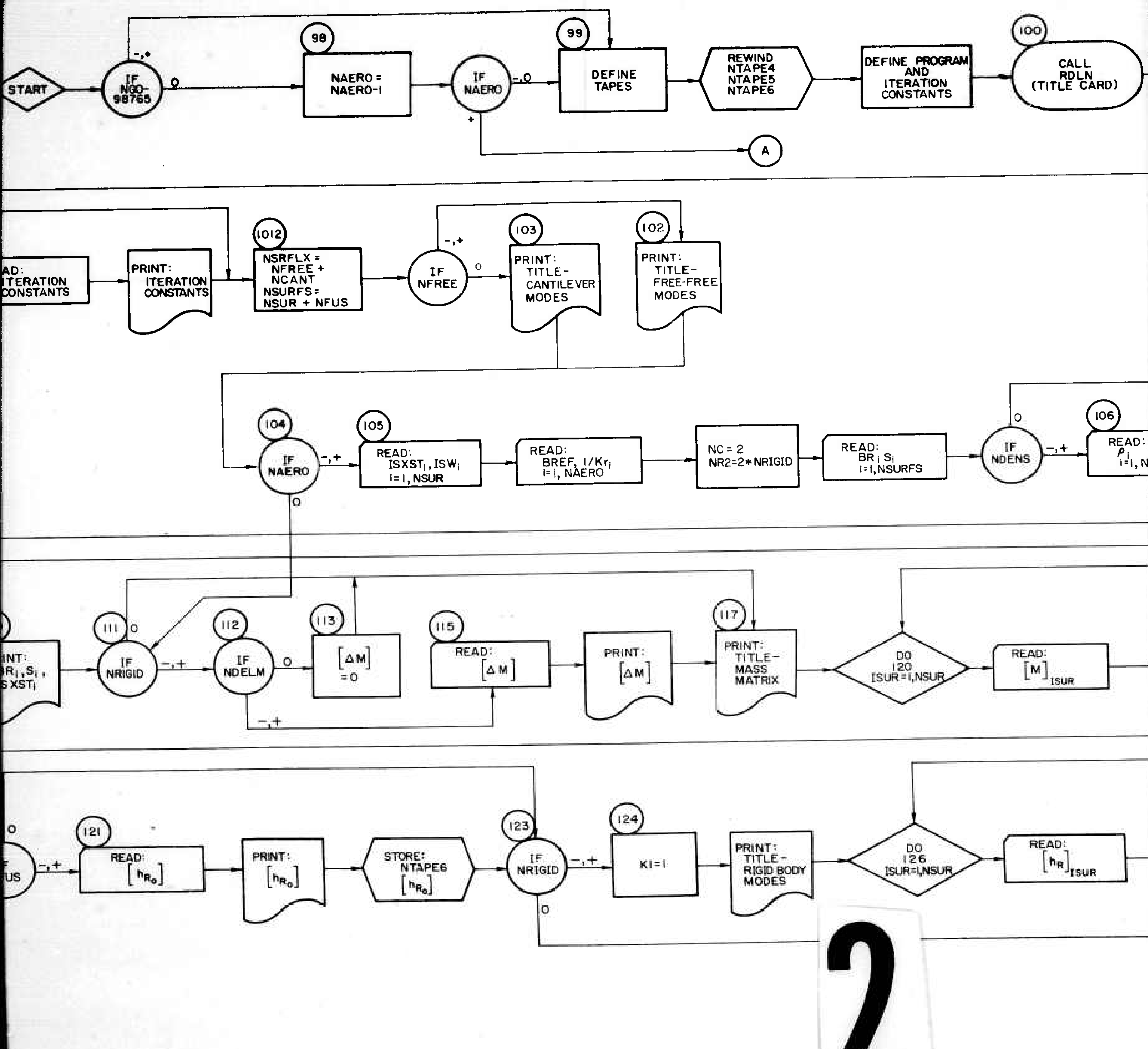
B. Generalized Tapes

The Aerospace IBM 7090 computer logical tape units used in the program are TAPES 2, 3, 5, 6, 7, 8, 11, 12, 13 and 14, but these may be altered by placing the desired machine logical tape units on symbolic cards HM140716 through HM140725. These symbolic cards appear in the main program, and the logical tape units are defined in the comments preceding the cards.


DIMENSIONS EQUIVALENCE COMMON FORMATS
START
IF NGO-98765
-,+
0
98
NAERO = NAERO-1
IF NAERO
-,0
+
99
IF NCON
0
-,+
1011
READ: ITERATION CONSTANTS
PRINT: ITERATION CONSTANTS
1012
NSRFLX = NFREE + NCANT
NSURFS = NSUR + NFUS
IF NFREE
-,+
0
103
PRINT: TITLE-CANTILEVE MODES
104
IF NAERO
-,+
0
105
READ: ISXST$_i$, ISW$_i$ I=1, NSUR
READ: BREF, I=1, NAE
109
PRINT: BREF
110
PRINT: BR$_i$, S$_i$, ISXST$_i$
111
IF NRIGID
0
-,+
112
IF NDELM
0
-,+
113
[ΔM] = 0
115
READ: [ΔM]
NPOINT= ΣSIZE ISUR
IF NFUS
0
-,+
121
READ: [h$_{R_0}$]
PRINT: [h$_{R_0}$]
STORE: NTAPE6 [h$_{R_0}$]
123
IF NRIGID
-,+
0

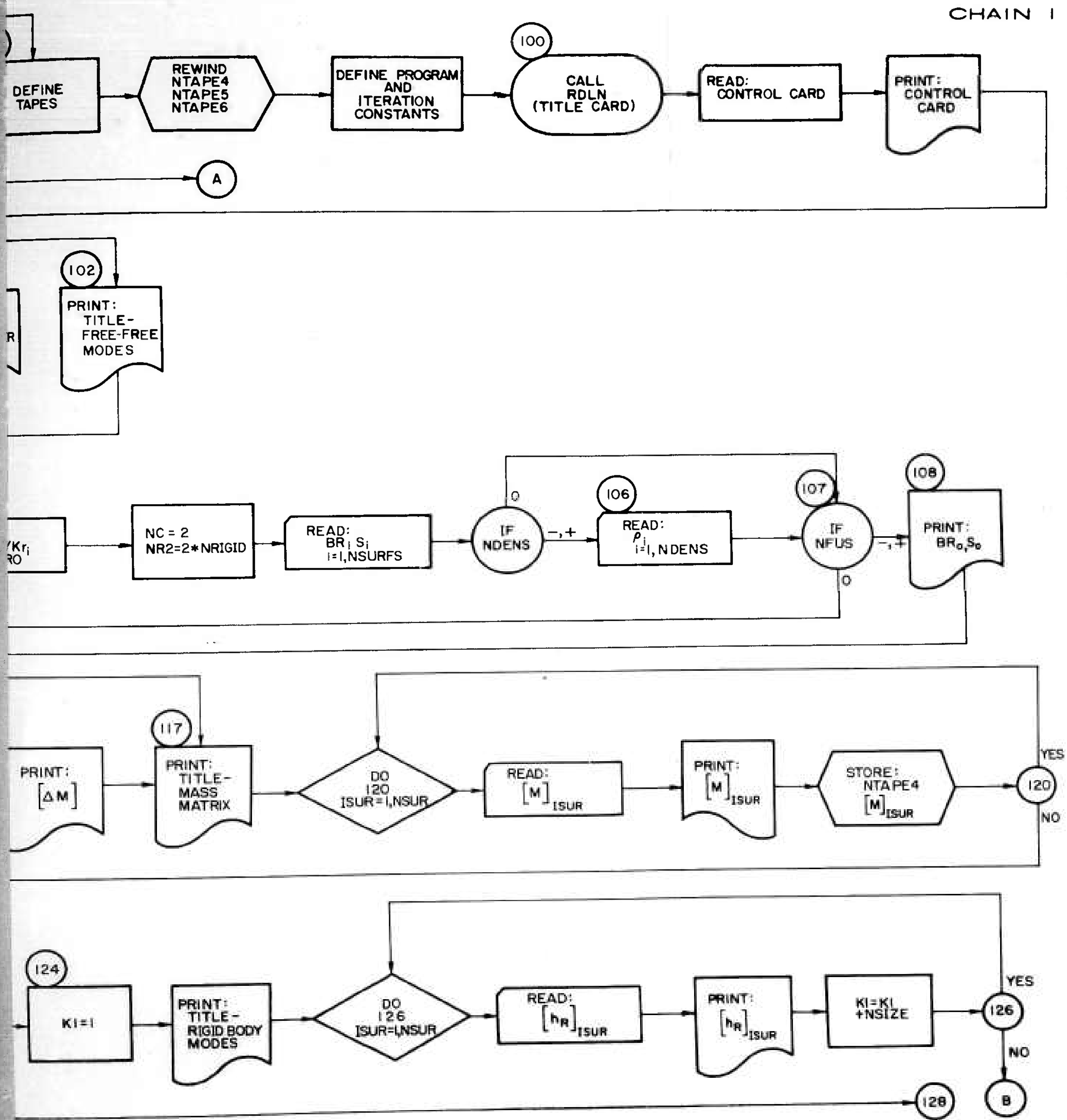
SECTION VII

FLOW DIAGRAM -- MAIN PROGRAM

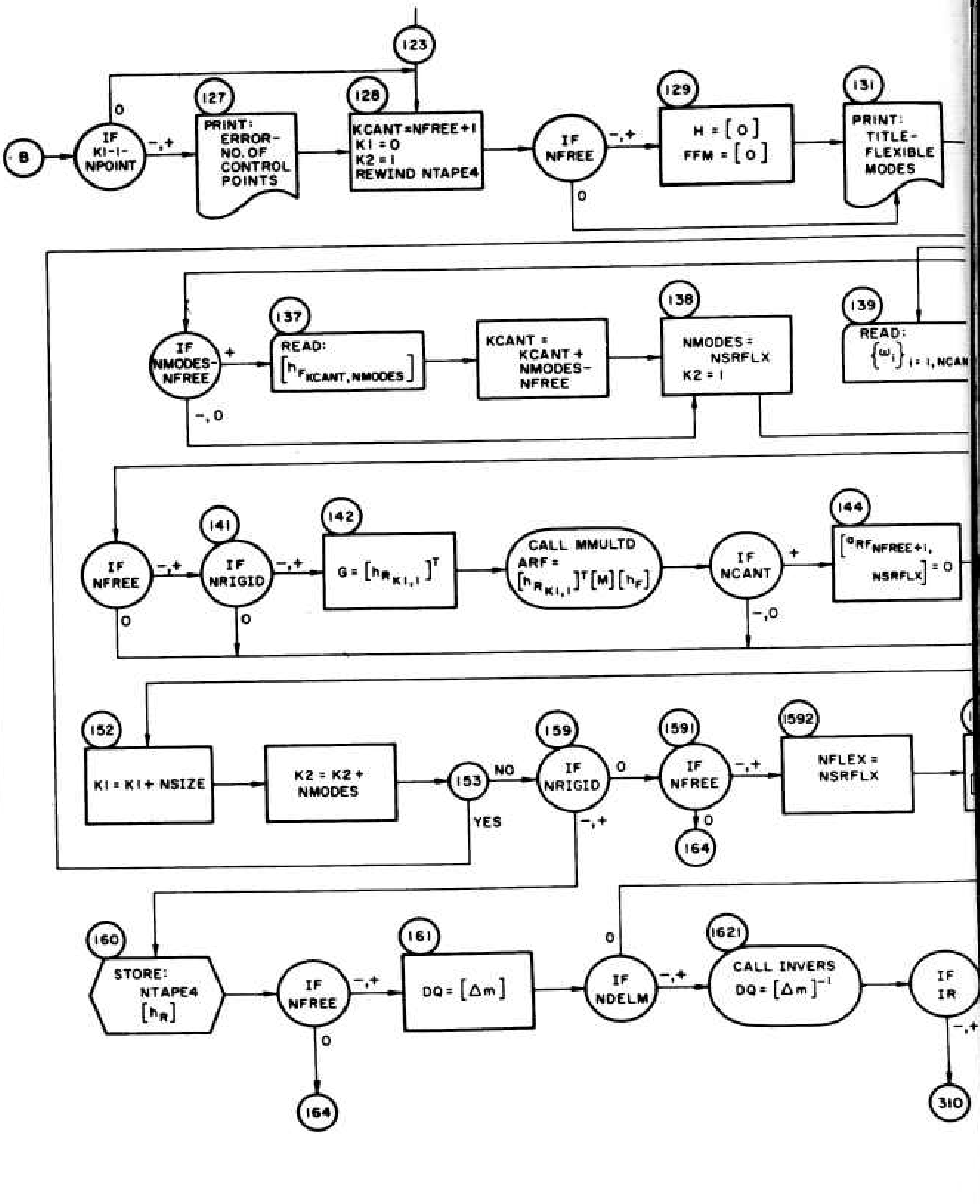


2

RAM -- MAIN PROGRAM

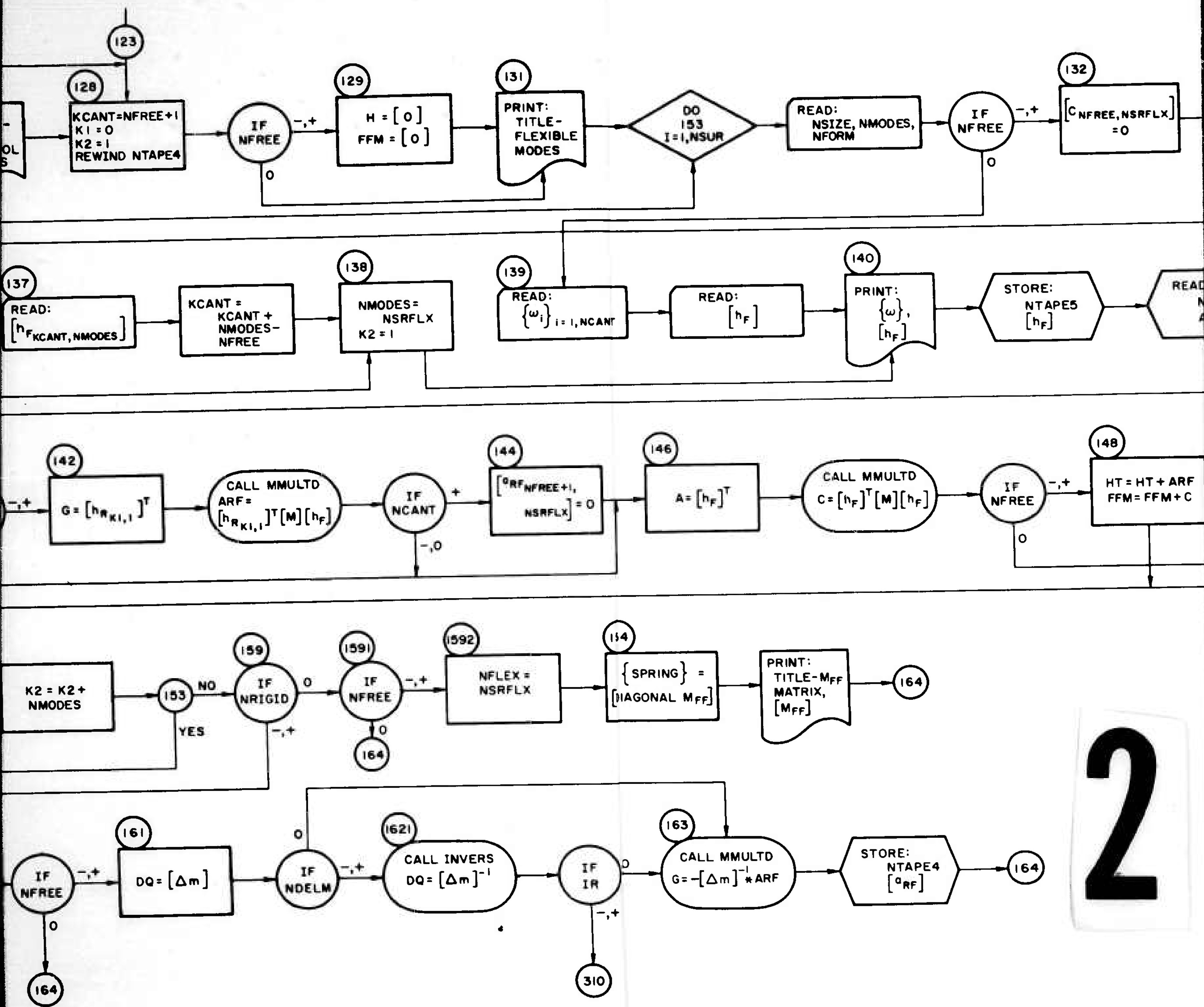


3



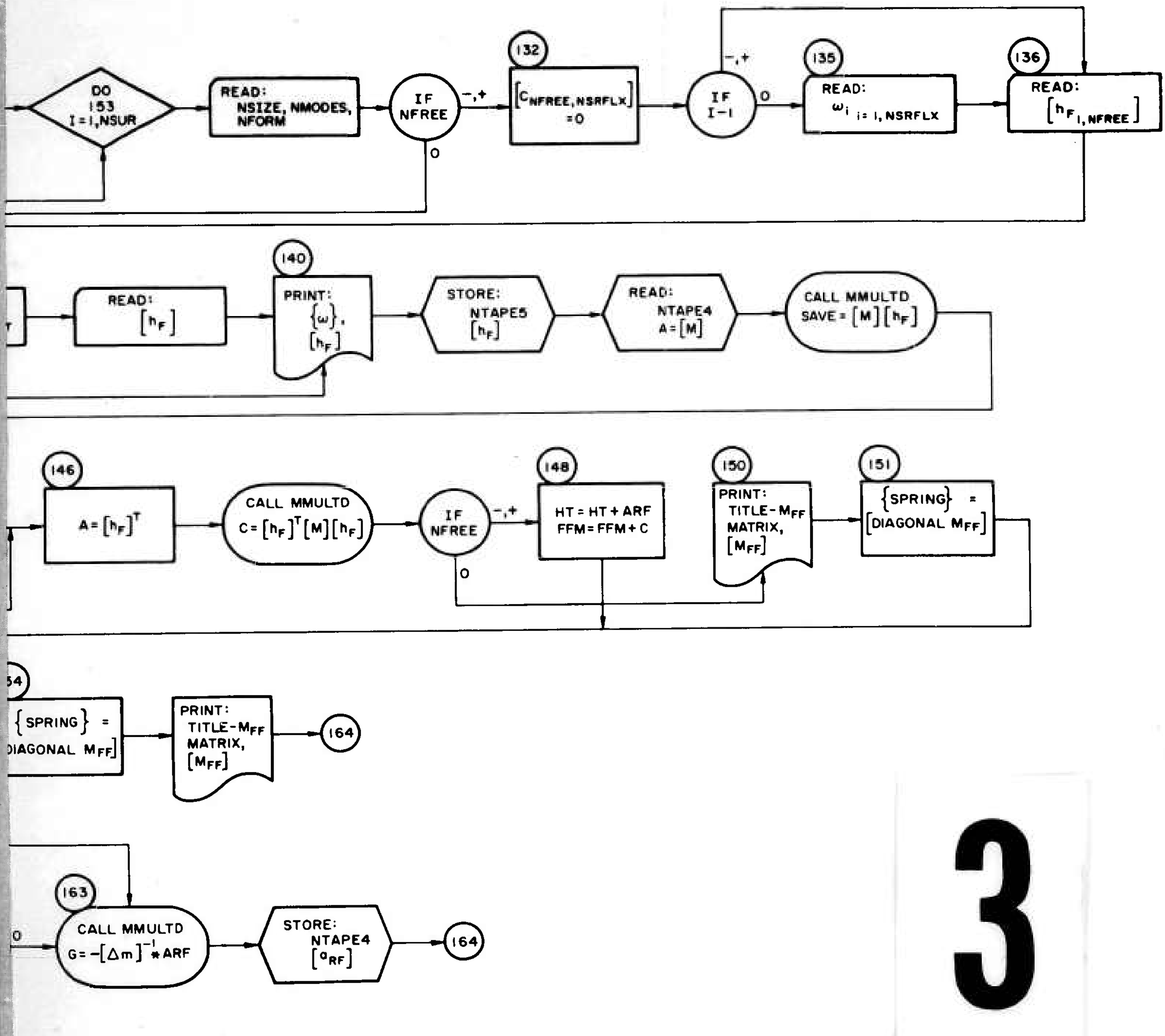
B
IF KI-I-NPOINT
0
-,+
127
PRINT: ERROR-NO. OF CONTROL POINTS
123
128
KCANT=NFREE+1
K1 = 0
K2 = 1
REWIND NTAPE4
IF NFREE
-,+
0
129
H = [0]
FFM = [0]
131
PRINT: TITLE-FLEXIBLE MODES
IF NMODES-NFREE
+
-,0
137
READ:
138
NMODES = NSRFLX
K2 = 1
KCANT = KCANT + NMODES-NFREE
139
READ:
IF NFREE
141
IF NRIGID
142
CALL MMULTD
IF NCANT
144
152
K1 = K1 + NSIZE
K2 = K2 + NMODES
153
NO
YES
159
IF NRIGID
1591
IF NFREE
1592
NFLEX = NSRFLX
164
160
STORE: NTAPE4
161
IF NFREE
IF NDELM
1621
CALL INVERS
IF IR
310

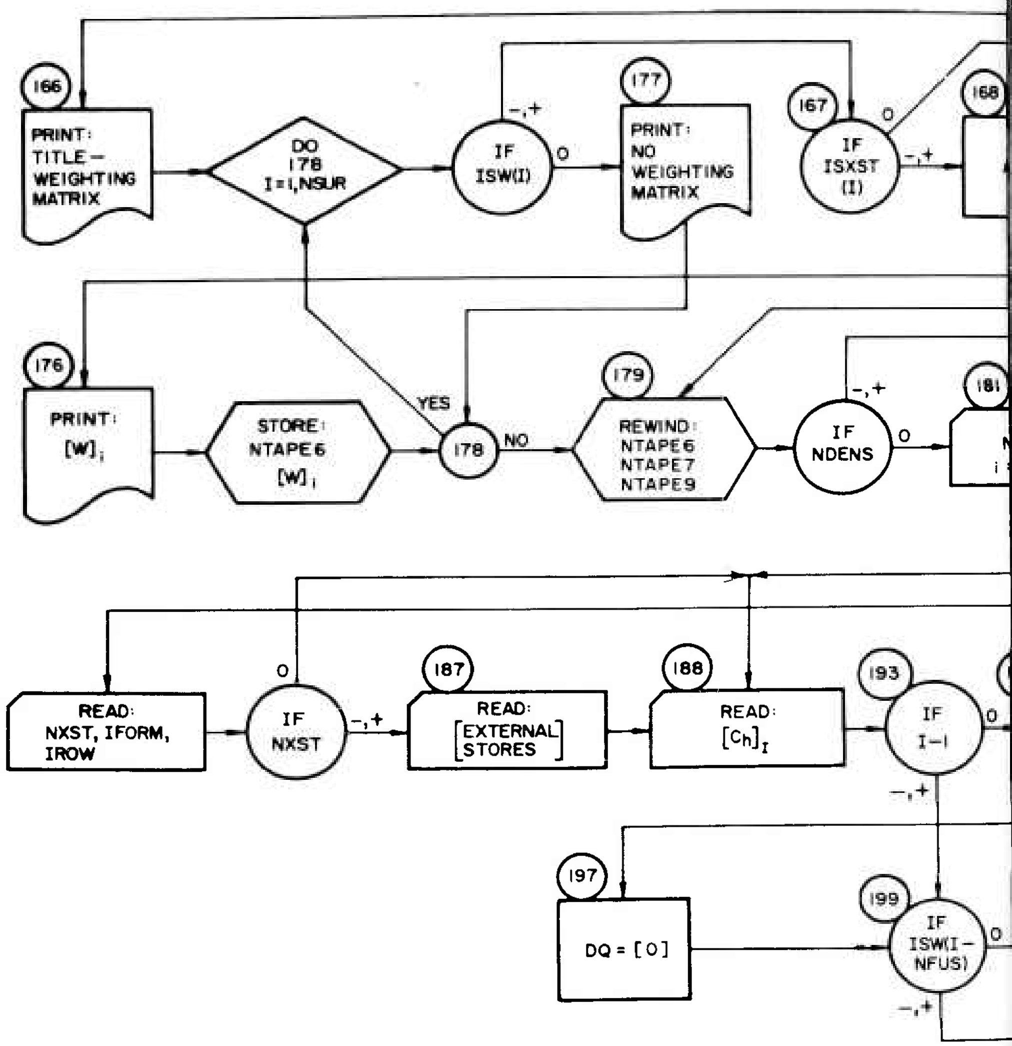
1



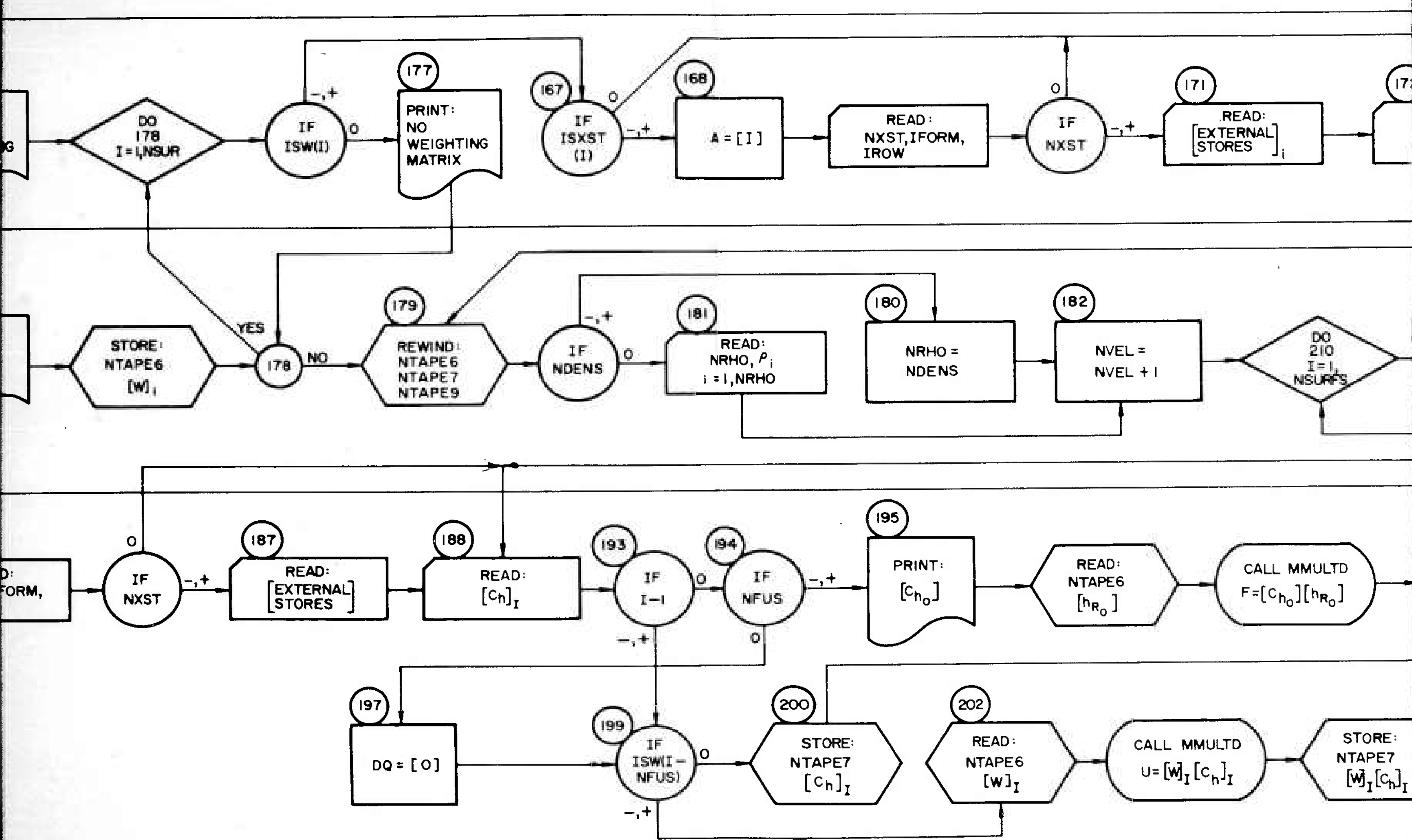
123
128
KCANT=NFREE+1
K1 = 0
K2 = 1
REWIND NTAPE4
IF NFREE
-,+
0
129
H = [0]
FFM = [0]
131
PRINT: TITLE-FLEXIBLE MODES
DO 153 I=1,NSUR
READ: NSIZE, NMODES, NFORM
IF NFREE
-,+
0
132
[C NFREE,NSRFLX] = 0
137
READ: [hF KCANT, NMODES]
KCANT = KCANT + NMODES- NFREE
138
NMODES= NSRFLX
K2 = 1
139
READ: {ωi} i=1,NCANT
READ: [hF]
140
PRINT: {ω}, [hF]
STORE: NTAPE5 [hF]
142
G = [hR KI,I]T
CALL MMULTD ARF= [hR KI,I]T [M] [hF]
IF NCANT
+
-,0
144
[aRF NFREE+1, NSRFLX] = 0
146
A = [hF]T
CALL MMULTD C = [hF]T [M] [hF]
IF NFREE
-,+
0
148
HT = HT + ARF
FFM = FFM + C
K2 = K2 + NMODES
153
NO
YES
159
IF NRIGID
0
-,+
1591
IF NFREE
-,+
0
164
1592
NFLEX = NSRFLX
154
{SPRING} = [DIAGONAL MFF]
PRINT: TITLE-MFF MATRIX, [MFF]
164
IF NFREE
-,+
0
164
161
DQ = [Δm]
IF NDELM
0
-,+
1621
CALL INVERS DQ = [Δm]-1
IF IR
0
-,+
310
163
CALL MMULTD G = -[Δm]-1 * ARF
STORE: NTAPE4 [aRF]
164

2

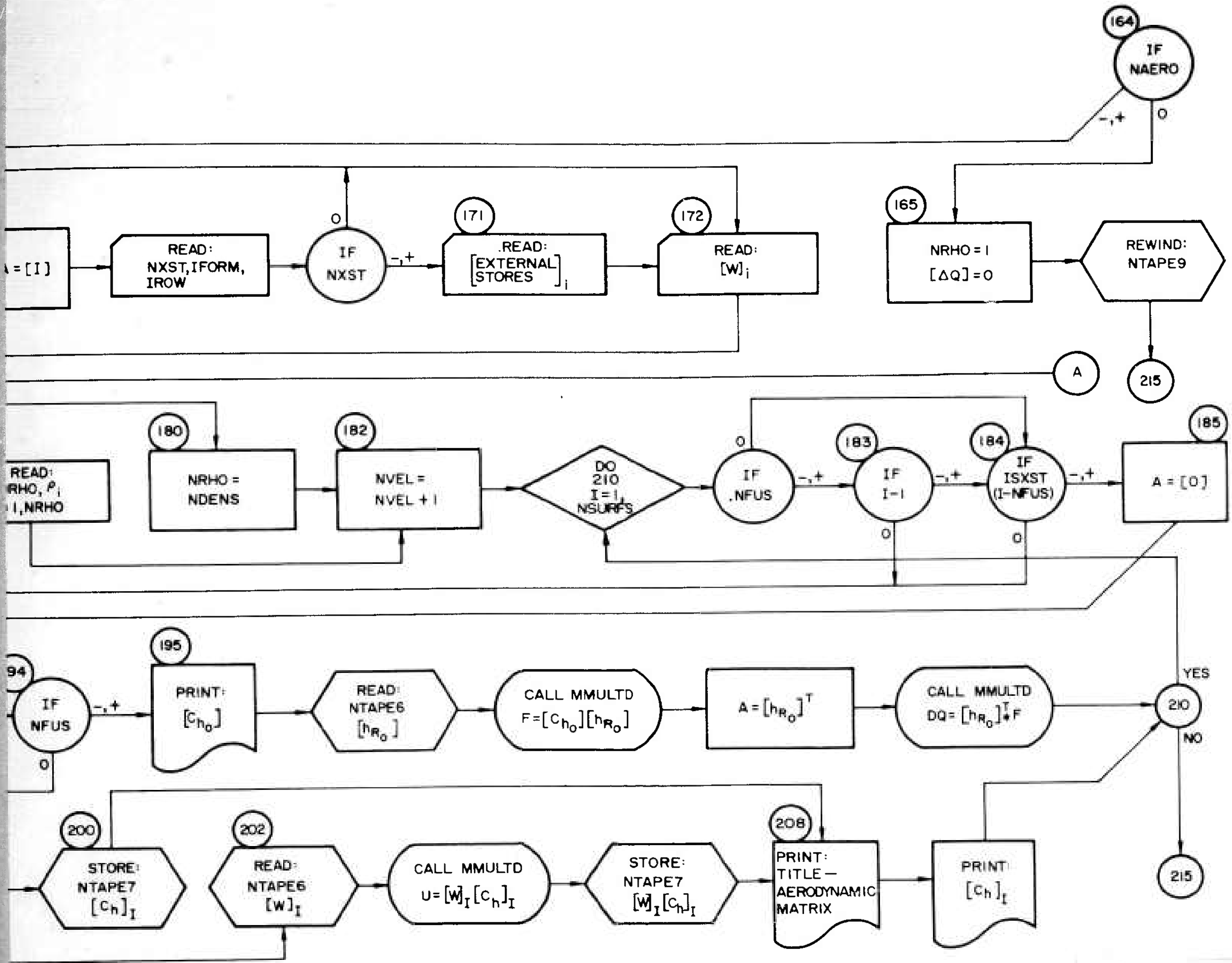

DO 153 I=1,NSUR
READ: NSIZE, NMODES, NFORM
IF NFREE
132
[C NFREE, NSRFLX] = 0
IF I-1
135
READ: ω i i=1, NSRFLX
136
READ: [h F 1, NFREE]
READ: [h F]
140
PRINT: {ω}, [h F]
STORE: NTAPE5 [h F]
READ: NTAPE4 A=[M]
CALL MMULTD SAVE=[M][h F]
146
A=[h F]T
CALL MMULTD C=[h F]T[M][h F]
IF NFREE
148
HT=HT+ARF FFM=FFM+C
150
PRINT: TITLE-MFF MATRIX, [MFF]
151
{SPRING} = [DIAGONAL MFF]
{SPRING} = [DIAGONAL MFF]
PRINT: TITLE-MFF MATRIX, [MFF]
164
163
CALL MMULTD G=-[Δm]-1*ARF
STORE: NTAPE4 [αRF]
164
3


166
PRINT: TITLE – WEIGHTING MATRIX
DO 178 I=1,NSUR
IF ISW(I)
-,+
0
177
PRINT: NO WEIGHTING MATRIX
167
IF ISXST (I)
0
-,+
168
176
PRINT: [W]$_i$
STORE: NTAPE6 [W]$_i$
YES
178
NO
179
REWIND: NTAPE6 NTAPE7 NTAPE9
IF NDENS
-,+
0
181
READ: NXST, IFORM, IROW
IF NXST
0
-,+
187
READ: [EXTERNAL STORES]
188
READ: [Ch]$_I$
193
IF I-1
0
-,+
197
DQ = [0]
199
IF ISW(I-NFUS)
0
-,+

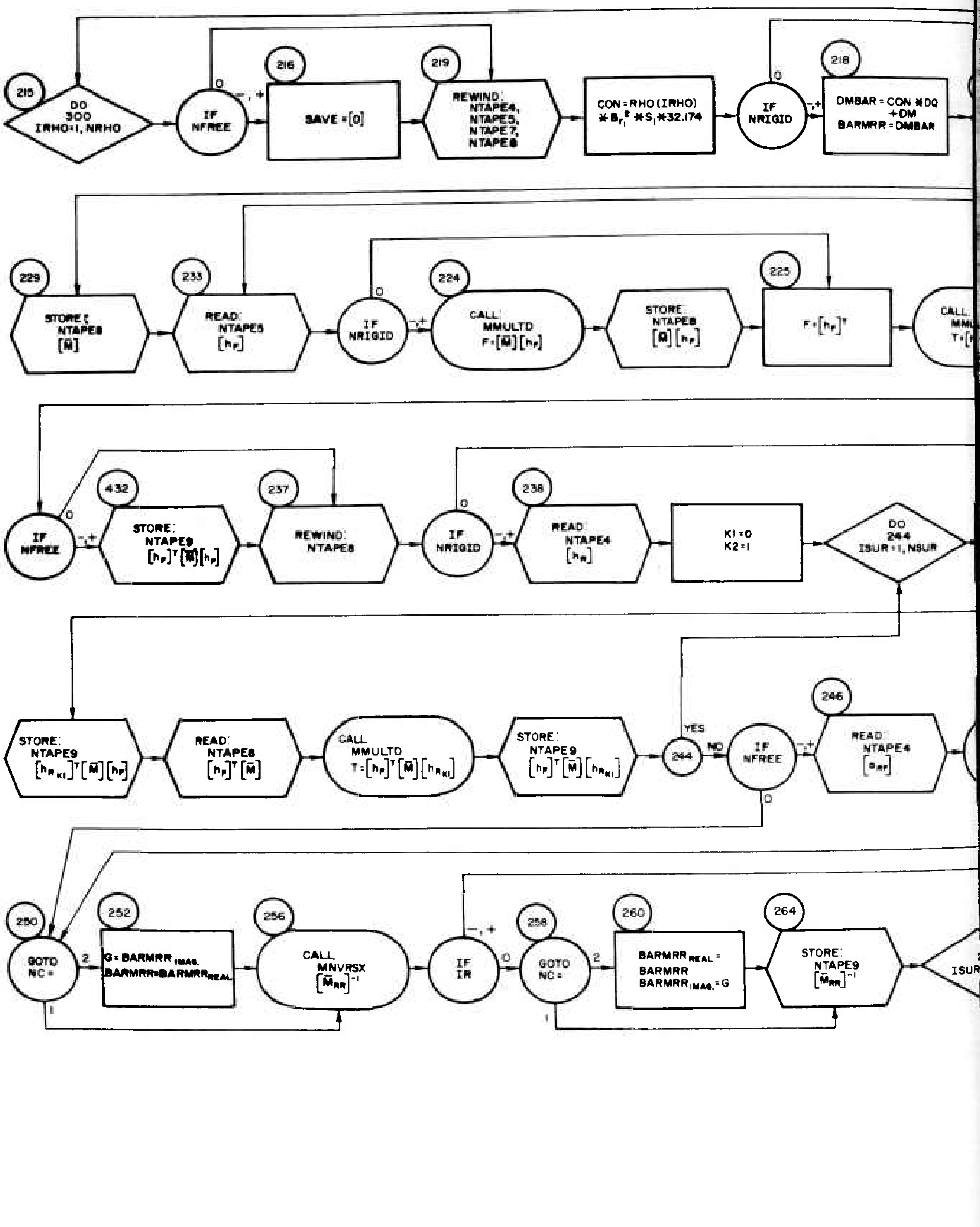
1


DO 178 I=1,NSUR
IF ISW(I)
-,+
0
177
PRINT: NO WEIGHTING MATRIX
167
IF ISXST (I)
0
-,+
168
A = [I]
READ: NXST,IFORM, IROW
IF NXST
0
-,+
171
READ: [EXTERNAL STORES]i
STORE: NTAPE6 [W]i
YES
178
NO
179
REWIND: NTAPE6 NTAPE7 NTAPE9
IF NDENS
-,+
0
181
READ: NRHO, ρi i=1,NRHO
180
NRHO = NDENS
182
NVEL = NVEL + 1
DO 210 I=1, NSURFS
0
IF NXST
-,+
187
READ: [EXTERNAL STORES]
188
READ: [Ch]I
193
IF I-1
0
-,+
194
IF NFUS
-,+
0
195
PRINT: [Cho]
READ: NTAPE6 [hRo]
CALL MMULTD F=[Cho][hRo]
197
DQ = [0]
199
IF ISW(I-NFUS)
0
-,+
200
STORE: NTAPE7 [Ch]I
202
READ: NTAPE6 [W]I
CALL MMULTD U=[W]I[Ch]I
STORE: NTAPE7 [W]I[Ch]I

2

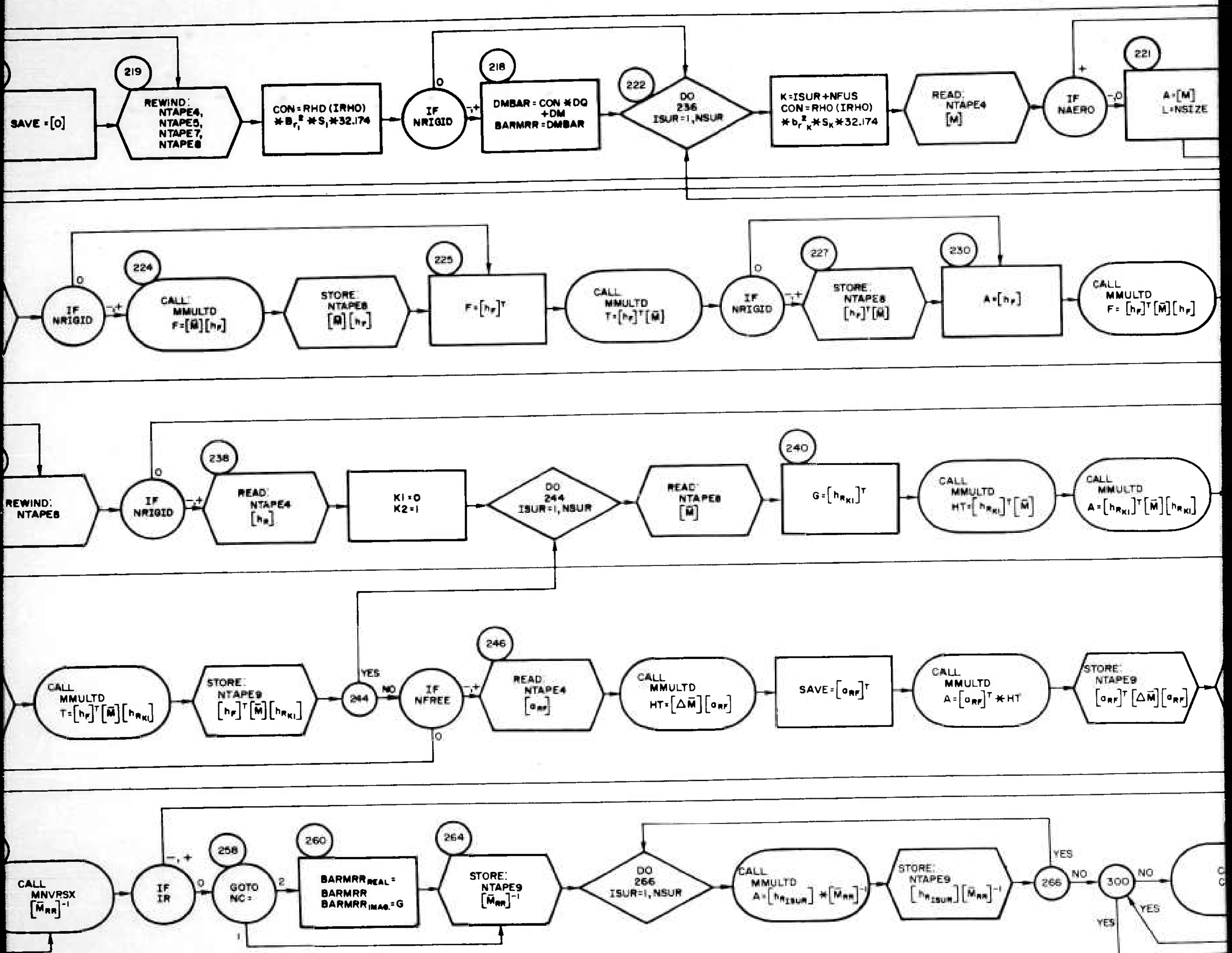

164
IF NAERO
-,+
0
165
NRHO = I
[ΔQ] = 0
REWIND: NTAPE9
215
A = [I]
READ: NXST, IFORM, IROW
IF NXST
0
-,+
171
READ: [EXTERNAL STORES]i
172
READ: [W]i
A
READ: NRHO, ρi I, NRHO
180
NRHO = NDENS
182
NVEL = NVEL + I
DO 210 I = I, NSURFS
IF NFUS
183
IF I-I
184
IF ISXST (I-NFUS)
185
A = [0]
194
IF NFUS
195
PRINT: [Cho]
READ: NTAPE6 [hRo]
CALL MMULTD F = [Cho][hRo]
A = [hRo]T
CALL MMULTD DQ = [hRo]T*F
210
YES
NO
215
200
STORE: NTAPE7 [Ch]I
202
READ: NTAPE6 [W]I
CALL MMULTD U = [W]I [Ch]I
STORE: NTAPE7 [W]I [Ch]I
208
PRINT: TITLE — AERODYNAMIC MATRIX
PRINT [Ch]I

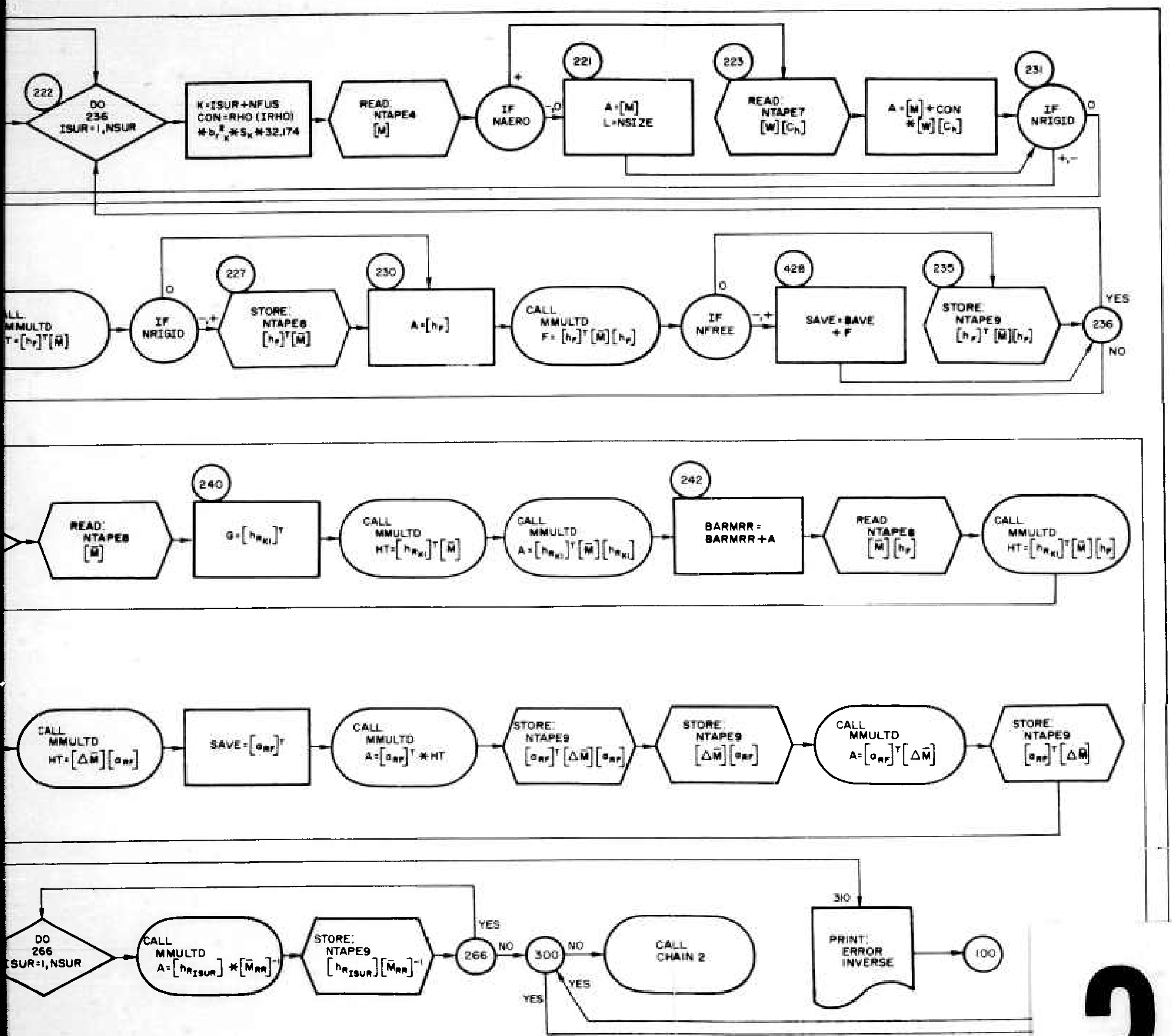
3

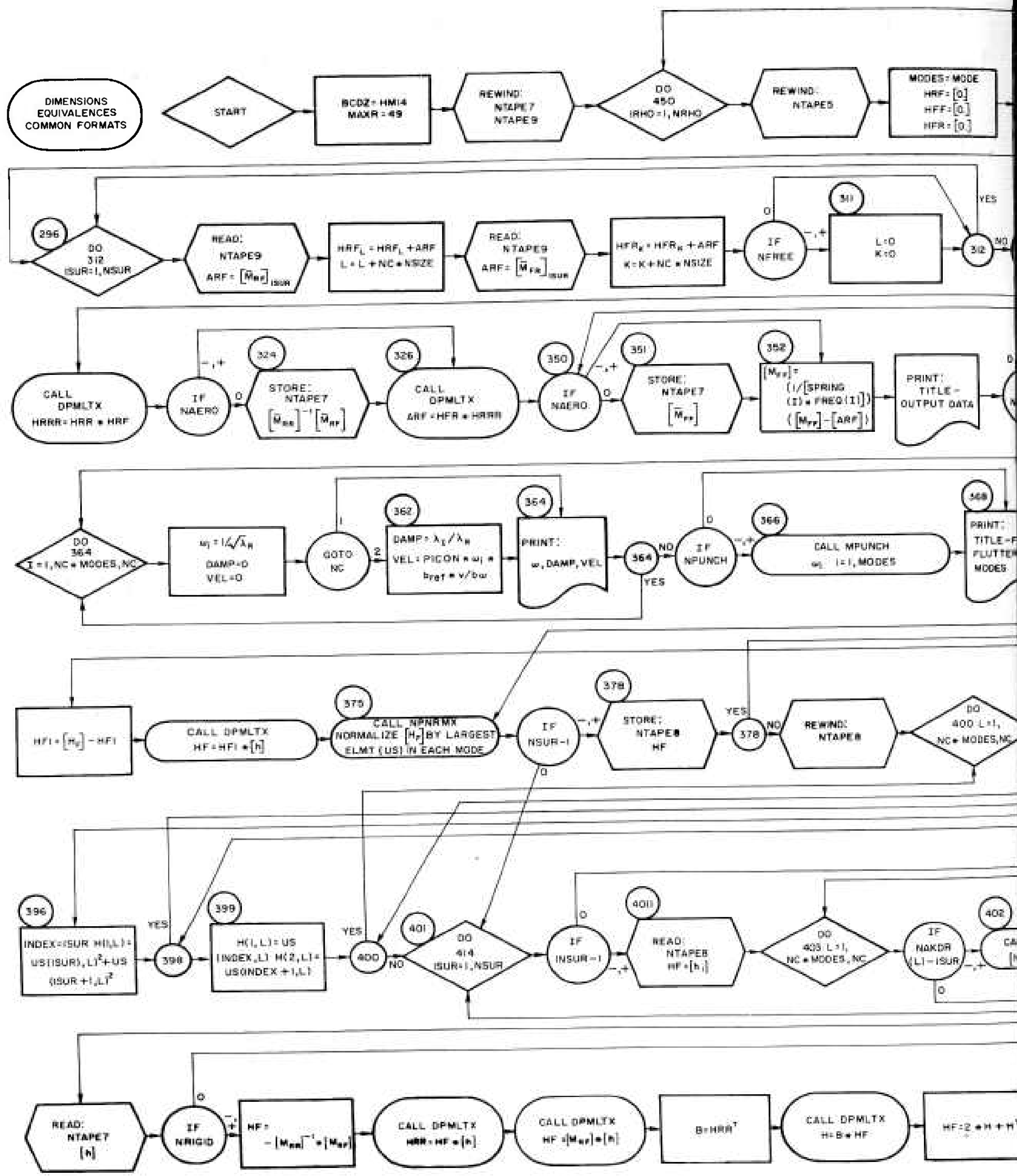


215
DO 300 IRHO=1, NRHO
IF NFREE
0
-,+
216
SAVE = [0]
219
REWIND: NTAPE4, NTAPE5, NTAPE7, NTAPE8
CON = RHO (IRHO) *B_r^2 *S_1 *32.174
IF NRIGID
218
DMBAR = CON *DQ +DM
BARMRR = DMBAR
229
STORE: NTAPE8 [M]
233
READ: NTAPE5 [h_F]
IF NRIGID
224
CALL MMULTD F=[M][h_F]
STORE: NTAPE8 [M][h_F]
225
F=[h_F]^T
CALL
IF NFREE
432
STORE: NTAPE9 [h_F]^T[M][h_F]
237
REWIND: NTAPE8
IF NRIGID
238
READ: NTAPE4 [h_R]
K1=0
K2=1
DO 244 ISUR=1, NSUR
STORE: NTAPE9 [h_RK1]^T[M][h_F]
READ: NTAPE8 [h_F]^T[M]
CALL MMULTD T=[h_F]^T[M][h_RK1]
STORE: NTAPE9 [h_F]^T[M][h_RK1]
YES
244
NO
IF NFREE
246
READ: NTAPE4 [a_RF]
250
GOTO NC=
2
1
252
G= BARMRR_IMAG.
BARMRR=BARMRR_REAL
256
CALL MNVRSX [M_RR]^-1
IF IR
258
GOTO NC=
260
BARMRR_REAL =
BARMRR
BARMRR_IMAG. = G
264
STORE: NTAPE9 [M_RR]^-1
ISUR

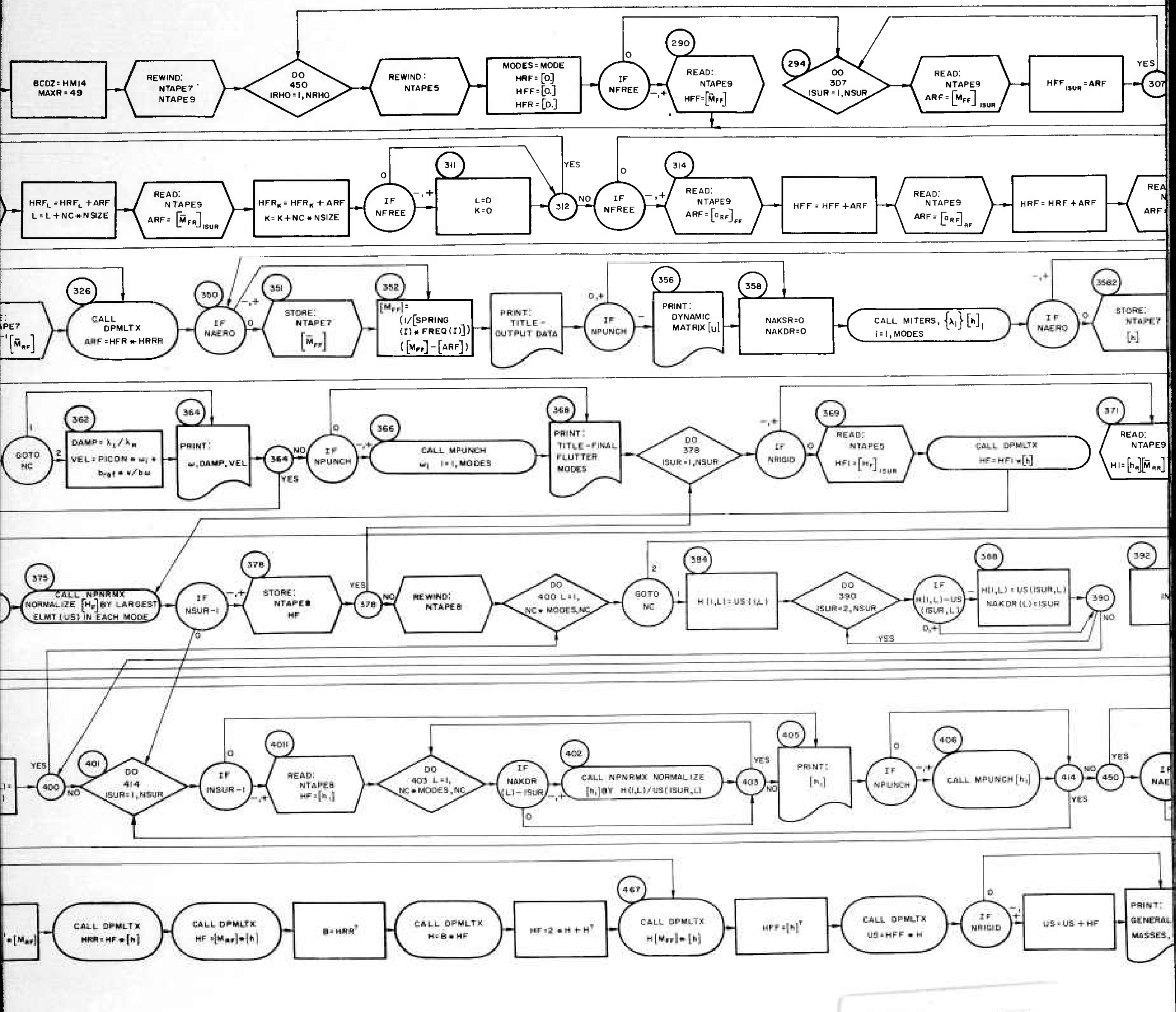
1

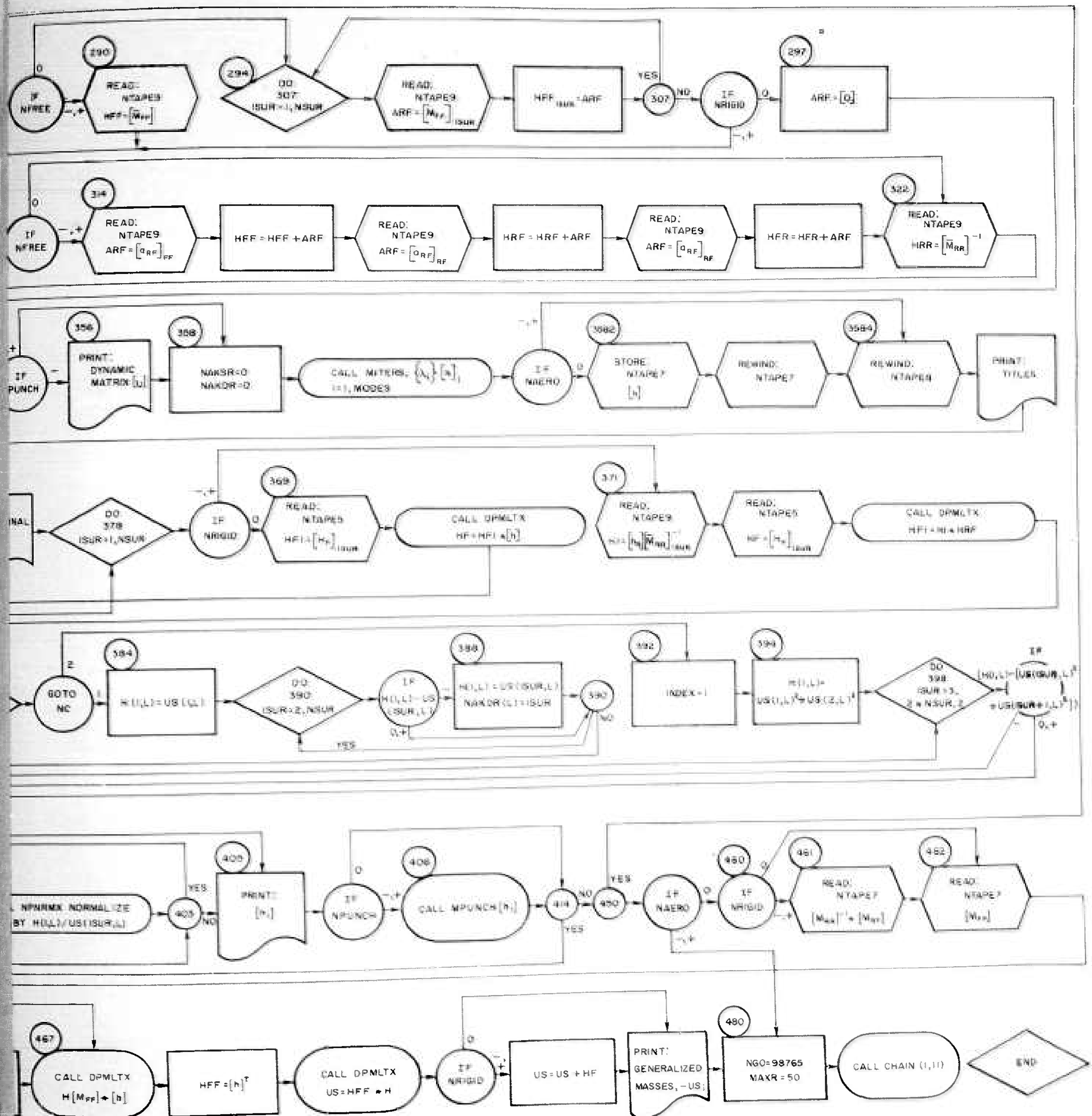

SAVE = [0]
219
REWIND: NTAPE4, NTAPE5, NTAPE7, NTAPE8
CON = RHO (IRHO) * B_r1^2 * S_1 * 32.174
IF NRIGID
218
DMBAR = CON * DQ + DM
BARMRR = DMBAR
222
DO 236 ISUR = 1, NSUR
K = ISUR + NFUS
CON = RHO (IRHO) * b_rK^2 * S_K * 32.174
READ: NTAPE4 [M]
IF NAERO
221
A = [M]
L = NSIZE
IF NRIGID
224
CALL MMULTD F = [M̄][h_F]
STORE: NTAPE8 [M̄][h_F]
225
F = [h_F]^T
CALL MMULTD T = [h_F]^T [M̄]
IF NRIGID
227
STORE: NTAPE8 [h_F]^T [M̄]
230
A = [h_F]
CALL MMULTD F = [h_F]^T [M̄] [h_F]
REWIND: NTAPE8
IF NRIGID
238
READ: NTAPE4 [h_R]
K1 = 0
K2 = 1
DO 244 ISUR = 1, NSUR
READ: NTAPE8 [M̄]
240
G = [h_RKI]^T
CALL MMULTD HT = [h_RKI]^T [M̄]
CALL MMULTD A = [h_RKI]^T [M̄] [h_RKI]
CALL MMULTD T = [h_F]^T [M̄] [h_RKI]
STORE: NTAPE9 [h_F]^T [M̄] [h_RKI]
YES
244
NO
IF NFREE
246
READ: NTAPE4 [a_RF]
CALL MMULTD HT = [ΔM̄] [a_RF]
SAVE = [a_RF]^T
CALL MMULTD A = [a_RF]^T * HT
STORE: NTAPE9 [a_RF]^T [ΔM̄] [a_RF]
CALL MNVRSX [M̄_RR]^-1
IF IR
258
GOTO NC =
260
BARMRR_REAL = BARMRR
BARMRR_IMAG = G
264
STORE: NTAPE9 [M̄_RR]^-1
DO 266 ISUR = 1, NSUR
CALL MMULTD A = [h_RISUR] * [M̄_RR]^-1
STORE: NTAPE9 [h_RISUR] [M̄_RR]^-1
YES
266
NO
300
NO
YES
YES


222
DO 236 ISUR=1,NSUR
K=ISUR+NFUS
CON=RHO (IRHO)
READ: NTAPE4 [M]
IF NAERO
221
A=[M] L=NSIZE
223
READ: NTAPE7 [W][C_h]
A=[M]+CON *[W][C_h]
231
IF NRIGID
CALL MMULTD
IF NRIGID
227
STORE: NTAPE8
230
A=[h_F]
CALL MMULTD
IF NFREE
428
SAVE=SAVE +F
235
STORE: NTAPE9
236
YES
NO
240
READ: NTAPE8 [M]
CALL MMULTD
CALL MMULTD
242
BARMRR= BARMRR+A
READ NTAPE8 [M][h_F]
CALL MMULTD
CALL MMULTD
SAVE=[a_RF]^T
CALL MMULTD
STORE: NTAPE9
STORE: NTAPE9
CALL MMULTD
STORE: NTAPE9
DO 266 ISUR=1,NSUR
CALL MMULTD
STORE: NTAPE9
266
300
CALL CHAIN 2
310
PRINT: ERROR INVERSE
100


DIMENSIONS EQUIVALENCES COMMON FORMATS
START
BCDZ = HM14 MAXR = 49
REWIND: NTAPE7 NTAPE9
DO 450 IRHO = 1, NRHO
REWIND: NTAPE5
MODES = MODE HRF = [0] HFF = [0] HFR = [0]
DO 312 ISUR = 1, NSUR
READ: NTAPE9 ARF = [M̄RF]ISUR
HRFL = HRFL + ARF L = L + NC * NSIZE
READ: NTAPE9 ARF = [M̄FR]ISUR
HFRK = HFRK + ARF K = K + NC * NSIZE
IF NFREE
L = 0 K = 0
CALL DPMLTX HRRR = HRR * HRF
IF NAERO
STORE: NTAPE7
CALL DPMLTX ARF = HFR * HRRR
IF NAERO
STORE: NTAPE7
PRINT: TITLE - OUTPUT DATA
DO 364 I = 1, NC * MODES, NC
DAMP = 0 VEL = 0
GOTO NC
PRINT: ω, DAMP, VEL
IF NPUNCH
CALL MPUNCH ωi I = 1, MODES
PRINT: TITLE - FLUTTER MODES
HF1 = [HF] - HF1
CALL DPMLTX HF = HF1 * [h]
CALL NPNRMX NORMALIZE [HF] BY LARGEST ELMT (US) IN EACH MODE
IF NSUR - 1
STORE: NTAPE8 HF
REWIND: NTAPE8
DO 400 L = 1, NC * MODES, NC
INDEX = ISUR H(I,L) = US(ISUR), L)² + US (ISUR + 1, L)²
H(1, L) = US (INDEX, L) H(2, L) = US(INDEX + 1, L)
DO 414 ISUR = 1, NSUR
IF INSUR - 1
READ: NTAPE8 HF = [hi]
DO 403 L = 1, NC * MODES, NC
IF NAKDR (L) - ISUR
READ: NTAPE7 [h]
IF NRIGID
CALL DPMLTX HRR = HF * [h]
CALL DPMLTX HF = [MRF] * [h]
B = HRR^T
CALL DPMLTX H = B * HF
HF = 2 * H + H^T
YES
NO
296
311
312
324
326
350
351
352
362
364
366
368
375
378
396
398
399
400
401
4011
402

1


BCDZ=HM14
MAXR=49
REWIND: NTAPE7 NTAPE9
DO 450 IRHO=1,NRHO
REWIND: NTAPE5
MODES=MODE HRF=[0.] HFF=[0.] HFR=[0.]
IF NFREE
READ: NTAPE9 HFF=[M̄FF]
DO 307 ISUR=1,NSUR
READ: NTAPE9 ARF=[MFF]ISUR
HFF ISUR=ARF
YES
HRFL=HRFL+ARF L=L+NC*NSIZE
READ: NTAPE9 ARF=[M̄FR]ISUR
HFRK=HFRK+ARF K=K+NC*NSIZE
IF NFREE
L=0 K=0
NO
IF NFREE
READ: NTAPE9 ARF=[0RF]FF
HFF=HFF+ARF
READ: NTAPE9 ARF=[0RF]RF
HRF=HRF+ARF
CALL DPMLTX ARF=HFR*HRRR
IF NAERO
STORE: NTAPE7 [M̄FF]
[MFF]=(1/[SPRING(I)*FREQ(I)])([MFF]-[ARF])
PRINT: TITLE-OUTPUT DATA
IF NPUNCH
PRINT: DYNAMIC MATRIX [U]
NAKSR=0 NAKDR=0
CALL MITERS, {λi}[h]i i=1,MODES
IF NAERO
STORE: NTAPE7 [h]
GOTO NC
DAMP=λI/λR VEL=PICON*ωi*bref*v/bω
PRINT: ω,DAMP,VEL
NO
YES
IF NPUNCH
CALL MPUNCH ωi i=1,MODES
PRINT: TITLE-FINAL FLUTTER MODES
DO 378 ISUR=1,NSUR
IF NRIGID
READ: NTAPE5 HFI=[HF]ISUR
CALL DPMLTX HF=HFI*[h]
READ: NTAPE9 HI=[hR][M̄RR]
CALL NPNRMX NORMALIZE [HF] BY LARGEST ELMT (US) IN EACH MODE
IF NSUR-1
STORE: NTAPE8 HF
REWIND: NTAPE8
DO 400 L=1, NC*MODES,NC
GOTO NC
H(I,L)=US(I,L)
DO 390 ISUR=2,NSUR
IF H(I,L)-US(ISUR,L)
H(I,L)=US(ISUR,L) NAKDR(L)=ISUR
DO 414 ISUR=1,NSUR
IF NSUR-1
READ: NTAPE8 HF=[hi]
DO 403 L=1, NC*MODES,NC
IF NAKDR(L)-ISUR
CALL NPNRMX NORMALIZE [hi] BY H(I,L)/US(ISUR,L)
PRINT: [hi]
IF NPUNCH
CALL MPUNCH [hi]
CALL DPMLTX HRR=HF*[h]
CALL DPMLTX HF=[MRF]*[h]
B=HRR^T
CALL DPMLTX H=B*HF
HF=2*H+H^T
CALL DPMLTX H[MFF]=[h]
HFF=[h]^T
CALL DPMLTX US=HFF*H
IF NRIGID
US=US+HF
PRINT: GENERAL MASSES,
2


IF NFREE
READ: NTAPE9 HFF = [M̄FF]
DO 307 ISUR = 1, NSUR
READ: NTAPE9 ARF = [MFF]ISUR
HFF ISUR = ARF
IF NRIGID
ARF = [0]
READ: NTAPE9 ARF = [aRF]FF
HFF = HFF + ARF
READ: NTAPE9 ARF = [aRF]RF
HRF = HRF + ARF
HFR = HFR + ARF
READ: NTAPE9 HRR = [M̄RR]^-1
IF PUNCH
PRINT: DYNAMIC MATRIX [D]
NAKSR = 0 NAKOR = 0
CALL MITERS, I = 1, MODES
IF NAERO
STORE: NTAPE7
REWIND: NTAPE7
REWIND: NTAPE9
PRINT: TITLES
DO 378 ISUR = 1, NSUR
READ: NTAPE5
CALL DPMLTX
GOTO NC
H (I,L) = US (I,L)
DO 390 ISUR = 2, NSUR
NAKOR (L) = ISUR
INDEX = I
NPNRMX NORMALIZE BY H(I,L)/US(ISUR,L)
PRINT: [hi]
IF NPUNCH
CALL MPUNCH [hi]
READ: NTAPE7
CALL DPMLTX H [MFF] * [h]
HFF = [h]^T
CALL DPMLTX US = HFF * H
US = US + HF
PRINT: GENERALIZED MASSES, - US;
NGO = 98765 MAXR = 50
CALL CHAIN (1,11)
END

3

SECTION VIII

SYMBOLIC LISTING

Some of the program FORTRAN symbols which were not defined in Section III, Part B are defined below.

FORTRAN Symbols	Definitions
BARMRR(I, J)	$[\overline{m}_{RR}]$ and $[\overline{m}_{RR}]^{-1}$
BOOLT	BCD titles
BREF	b_r (reference)
BR(I)	b_{ri} series
DM(I, J)	$[\Delta m]$
DMBAR(I, J)	$[\Delta \overline{m}]$
DQ(I, J)	$[h_{Ro}]^T [C_{ho}] [h_{Ro}]$
MAXS, MAXR, MAXP, MAXQ	Dimensional number of rows in working matrices
NFLEX, NFLEX2	Total number of flexible modes
NGO	Chain 1 entry control number
NPOINT	Total number of control points on all flexible surfaces
NRIGID, NR2	Total number of rigid-body modes
NSIZES(I)	Number of control points on surface i
NSRFLEX	NFREE + NCANT
NSURFS	NSUR + NFUS, (NFUS = 0 or 1)
PICON	$2 \times \pi \times 0.5921$
RHO(I)	ρ_i, i = I, NDENS

SPRING	$\left[K/4\pi^2\right]$
T, G, A, SAVE, H, ARF, C, FFM, F, HR, HT	Working areas of core
VELCTY(I)	$(1/k_r)_i = (V/b_r\omega)_i$

The complete symbolic listing is given on the following pages.

```
                                                                        HM140619
C     NSUR=TOTAL NUMBER OF SURFACES ALLOWED.                            HM140620
C     NDENS=TOTAL NUMBER OF DENSITIES ALLOWED.                          HM140621
C     NRIGID=TOTAL NUMBER OF RIGID BODIES ALLOWED                       HM140622
C     NSIZE=TOTAL NUMBER CONTROL POINTS ON ANY ONE SURFACE ALLOWED      HM140623
C     NMODES=TOTAL NUMBER MODES INPUT ON ANY ONE SURFACE.               HM140624
C     DIMENSION  ISXST(NSUR), ISW(NSUR), RHO(NDENS), BR(NSUR+1),        HM140625
C    1           S(NSUR+1), DM(NRIGID),LOW(NSIZE), LHIGH(NSIZE),        HM140626
C    2           HR(NSUR*NSIZE,NRIGID), FREQ(NSUR*NMODES),              HM140627
C    3           DM(NRIGID,NRIGID), DMBAR(NRIGID,2*NRIGID), DQ(NRIGID,  HM140628
C    4           2*NRIGID)                                              HM140629
                                                                        HM140630
      DIMENSION  ISXST(20), ISW(20), BR(21), S(21), RHO(20), DM(6,6),   HM140631
     1           DQ(6,12), DMBAR(6,12), BARMRR(6,12), FREQ(40),         HM140632
     2           LOW(50), LHIGH(50),          NSIZES(20),  IT(520),     HM140633
     3           A(50,100), C(50,50), HR(1000,6),F(50,100), G(6,50),    HM140634
     4           HT(6,100), H(50,12), ARF(6,40), T(50,100)              HM140635
     5           ,SPRING(40),SAVE(50,50),VELCTY(20) ,FFM(50,50)         HM140636
                                                                        HM140637
      EQUIVALENCE  (IT(1),ISXST(1)), (IT(21),ISW(1)), (IT(41),NSIZES)   HM140638
     1            ,(IT(61),NTAPE1), (IT(62),NTAPE2), (IT(63),NTAPE3),   HM140639
     2             (IT(64),NTAPE4), (IT(65),NTAPE5), (IT(66),NTAPE6),   HM140640
     3             (IT(67),NTAPE7), (IT(68),NTAPE8), (IT(69),NTAPE9),   HM140641
     4             (IT(70),NSUR), (IT(71),NRIGID), (IT(72),NFREE),      HM140642
     5             (IT(73),NCANT), (IT(74),NFUS), (IT(75),NDENS),       HM140643
     6             (IT(76),MODES), (IT(77),NPOINT), (IT(78),NFLEX),     HM140644
     7             (IT(79),MAXR), (IT(80),MAXQ), (IT(81),MAXS),         HM140645
     8             (IT(82),MAXP), (IT(83),NSURFS), (IT(84),NR2),        HM140646
     9             (IT(85),NFLEX2), (IT(86),NC), (IT(87),NGO)           HM140647
      EQUIVALENCE  (IT(88),NPUNCH), (IT(89),BDOLT), (IT(91),RHO),       HM140648
     1             (IT(111),FREQ), (IT(151),BREF), (IT(152),NSRFLX),    HM140649
     2             (IT(153),BR), (IT(174),S), (IT(195),DM), (IT(231),   HM140650
     3            DQ), (IT(303), DMBAR), (IT(375),BARMRR),              HM140651
     4           (IT(447),NAERO),  (IT(448),NRHO), (IT(449),PICON),     HM140652
     5             (IT(450),SPRING), (IT(490),VELCTY),(IT(511),NVEL),   HM140653
     6             (IT(512),NTAPE0), (IT(513),NCARDS), (IT(514),EPSP),  HM140654
     7             (IT(515),EPDP), (IT(516),NITRSP), (IT(517),NITRDP),  HM140655
```

```
 8            (IT(518),AITKEN), (A(2501),C), (HR,F), (ARF,T,LOW),  HM140656
 9            (T(241),HT,H), (T(841),G,LHIGH)                      HM140657
                                                                   HM140658
  COMMON     IT, A, HR, T ,SAVE,FFM                                HM140659
                                                                   HM140660
 1 FORMAT  (18I4)                                                  HM140661
 2 FORMAT  (6E12.8)                                                HM140662
 3 FORMAT  (1H   16X,  41H FLUTTER ANALYSIS BY A MOOAL METHOO USING HM140663
 1         35H AEROOYNAMIC INFLUENCE COEFFICIENTS   /// 10HO  NSUR = HM140664
 2         1I2, 10H  NAERO =  1I4, 11H  NRIGID = 1I2, 9H  NFUS =   HM140665
 3         1I2, 10H  NOENS =  1I4, 14H  MODES OUT =  1I2,  7H  NFREE HM140666
 4         3H =  1I3,  10H  NCANT =  1I3  )                        HM140667
 4 FORMAT  (1HO  38X,  11H B (REF) =   1E20.8,  / 1HO 20X, 7HSURFACE HM140668
 1          17X, 2H B  18X, 2H S  10X, 20HEXTERNAL STORES SIZE  ///) HM140669
 5 FORMAT  (1HO 10X, 21H B RIGIO COMPONENT =  1E18.8, 5X, 8H S RIGIO HM140670
 1          13H COMPONENT =     1E18.8     )                       HM140671
 6 FORMAT  (1H  1I25,  2(  5X,  1E20.8),  1I12 )                   HM140672
 7 FORMAT  (1HO 20X, 23H FREE-FREE INPUT MODES,  1I6, 11H CANTILEVER HM140673
 1         16H MODES INCLUOED.  )                                  HM140674
 8 FORMAT  (1HO  28X, 24H CANTILEVER INPUT MOOES     )             HM140675
 9 FORMAT  (1HO  45X, 1I6, 13H FREQUIENCIES  //  (1H  45X, 1E 20.8) )HM140676
10 FORMAT  (1H1  48X, 12H MASS MATRIX    )                         HM140677
11 FORMAT  (41HO NUMBER OF CONTROL POINTS THIS MATRIX, ( 1I4,      HM140678
 1          48H) AND TOTAL NUMBER OF CONTROL POINTS EXPECTEO, (  1I4,HM140679
 2          37H) OO NOT AGREE, PROGRAM CONTINUEO....    )          HM140680
12 FORMAT  (1H1 42X,24H RIGID BODY MOOAL MATRIX    )               HM140681
13 FORMAT  (1H  38X,  8HSURFACE  1I2,  1H,  1I6, 15H CONTROL POINTS) HM140682
14 FORMAT  (1H1  43X,  23H FLEXIBLE MODAL MATRIX  )                 HM140683
15 FORMAT  (1H1  46X, 18H WEIGHTING MATRIX   )                     HM140684
16 FORMAT  (1H1 28X, 29H RIGIO COMPONENT AERO MATRIX, 1I9,8H CONTROL HM140685
 1          7H POINTS  )                                           HM140686
17 FORMAT  (1H1    33X,  20H AERODYNAMIC MATRIX  3X,        5X,    HM140687
 1                10H 1./K R =      1E20.8  )                      HM140688
18 FORMAT  (1HO 30X, 23H RIGIO COMPONENT MOOES,   1I10,            HM140689
 1          17H CONTROL POINTS.   )                                HM140690
19 FORMAT  (1HO 30X, 29H RIGIO COMPONENT MASS MATRIX      )         HM140691
20 FORMAT  (47H1 ERROR IN INVERSE ROUTINE, PROGRAM TERMINATEO    )  HM140692
21 FORMAT  ( 1H   38X, 8HSURFACE  1I2,  1H, 6X, 13H NO WEIGHTING   HM140693
```

```
   1          7H MATRIX  )                                                HM140694
  22 FORMAT (1H0  40X, 21H M FF MATRIX,SURFACE   1I6   )                  HM140695
  23 FORMAT  ( 3E12.8,  2I4 )                                             HM140696
  24 FORMAT  ( 8H0EPSP =   1E16.8, 1X, 8H EPDP =  1E16.8, 1X,             HM140697
   1          10H AITKEN =  1E16.8, 1X, 10H NITRSP = 1I4, 1X,             HM140698
   2          10H NITRDP =   1I4  )                                       HM140699
                                                                          HM140700
C     NTAPE0 = OUTPUT PUNCH TAPE                                          HM140701
C     NTAPE1 = CHAIN TAPE                                                 HM140702
C     NTAPE2 = INPUT TAPE                                                 HM140703
C     NTAPE3 = OUTPUT PRINT TAPE                                          HM140704
C     NTAPE4 =     /                                                      HM140705
C     NTAPE5 =    /   ARE UTILITY TAPES                                   HM140706
C     NTAPE6 =   /                                                        HM140707
C     NTAPE7 = /                                                          HM140708
C     NTAPE8 =/                                                           HM140709
C     NTAPE9=/                                                            HM140710
                                                                          HM140711
      IF ( NGO-98765  )       99,98,99                                    HM140712
  98 NAERO = NAERO-1                                                      HM140713
      IF  ( NAERO )           99,99,179                                   HM140714
                                                                          HM140715
  99 NTAPE1 = 11                                                          HM140716
     NTAPE2 = 2                                                           HM140717
     NTAPE3 = 3                                                           HM140718
     NTAPE4 = 13                                                          HM140719
     NTAPE5 = 5                                                           HM140720
     NTAPE6 = 6                                                           HM140721
     NTAPE7 = 7                                                           HM140722
     NTAPE8 = 8                                                           HM140723
     NTAPE9 = 14                                                          HM140724
     NTAPE0 = 12                                                          HM140725
     CALL SETH(NTAPE5,NTAPE6,NTAPE7,NTAPE8,NTAPE4,NTAPE9)                 HM140726
                                                                          HM140727
     MAXR = 50                                                            HM140728
     MAXQ = 6                                                             HM140729
     MAXS = 1000                                                          HM140730
     MAXP = 40                                                            HM140731
```

```
      NVEL=0                                                            HM140732
      NCARDS = 0                                                        HM140733
      EPSP = .5E-06                                                     HM140734
      EPDP = .5E-07                                                     HM140735
      AITKEN = .9                                                       HM140736
      NITRSP = 40                                                       HM140737
      NITRDP = 100                                                      HM140738
      PICON=.5921*2.*3.14159                                            HM140739
      NC=1                                                              HM140740
                                                                        HM140741
      REWIND NTAPE4                                                     HM140742
      REWIND NTAPE5                                                     HM140743
      REWIND NTAPE6                                                     HM140744
                                                                        HM140745
C***********************************************************************HM140746
C READ IN TITLE, CONTROLS AND CONSTANTS AND PRINT.                       HM140747
                                                                        HM140748
  100 CALL RDLN  (NTAPE2,NTAPE3,1)                                      HM140749
      READ INPUT TAPE NTAPE2, 1, NSUR, NAERO, NRIGID, NFUS, NDENS, MODESHM140750
     1                       ,NFREE, NCANT, NDELM, NPUNCH, NCON         HM140751
      WRITE OUTPUT TAPE NTAPE3, 3, NSUR, NAERO, NRIGID, NFUS, NDENS,    HM140752
     1                         MODES, NFREE, NCANT                      HM140753
      IF  ( NCON )          1011,1012,1011                              HM140754
 1011 READ INPUT TAPE NTAPE2, 23, EPSP, EPDP, AITKEN, NITRSP, NITRDP    HM140755
      WRITE OUTPUT TAPE NTAPE3, 24, EPSP, EPDP, AITKEN, NITRSP, NITRDP  HM140756
 1012 NSRFLX=NFREE+NCANT                                                HM140757
      NSURFS=NSUR+NFUS                                                  HM140758
      NR2=  NRIGID *NC                                                  HM140759
      IF ( NFREE )          102,103,102                                 HM140760
  102 WRITE OUTPUT TAPE NTAPE3,  7,  NCANT                              HM140761
      GOTO  104                                                         HM140762
  103 WRITE OUTPUT TAPE NTAPE3,  8                                      HM140763
  104 IF (NAERO)            105,111,105                                 HM140764
  105 READ INPUT TAPE NTAPE2, 1,  (ISXST(I),ISW(I),I=1,NSUR)            HM140765
      READ INPUT TAPE NTAPE2, 2, BREF,  (VELCTY(I),I=1,NAERO)           HM140766
      NC=2                                                              HM140767
      NR2=  NRIGID *NC                                                  HM140768
      READ INPUT TAPE NTAPE2, 2,  (BR(I),S(I),I=1,NSURFS)               HM140769
```

```
      IF ( NDENS )          106,107,106                                 HM140770
  106 READ INPUT TAPE NTAPE2, 2,  (RHO(I),I=1,NDENS)                    HM140771
                                                                        HM140772
  107 IF ( NFUS )           108,109,108                                 HM140773
  108 WRITE OUTPUT TAPE NTAPE3, 5, BR(1), S(1)                          HM140774
                                                                        HM140775
  109 WRITE OUTPUT TAPE NTAPE3, 4, 8REF                                 HM140776
      DO  110 I=1,NSUR                                                  HM140777
        J=I+NFUS                                                        HM140778
  110   WRITE OUTPUT TAPE NTAPE3, 6, (I, BR(J), S(J) ,ISXST(I))         HM140779
                                                                        HM140780
C***********************************************************************HM140781
C READ IN FUSELAGE MASS CHARACTERISTICS                                  HM140782
  111 IF  ( NRIGID )         112,117,112                                HM140783
  112 IF  ( NOELM  )         115,113,115                                HM140784
  113 00  114 I=1,NRIGID                                                HM140785
        DO   114 J=1,NRIGIO                                             HM140786
  114     DM(I,J)=0.                                                    HM140787
      GOTU  117                                                         HM140788
  115 OD  116  I=1,NRIGIO                                               HM140789
  116   READ INPUT TAPE NTAPE2, 2,  (DM(J,I),J=1,NRIGID)                HM140790
      WRITE DUTPUT TAPE NTAPE3, 19,                                     HM140791
      CALL MPRINT (DM,NRIGID,NRIGID,MAXQ,NTAPE3)                        HM140792
                                                                        HM140793
C***********************************************************************HM140794
C REAO MASS MATRIX FOR EACH SURFACE, STORE SYSTEM MASS MATRIX DN NTAPE4 HM140795
                                                                        HM140796
  117 K1=0                                                              HM140797
        WRITE OUTPUT TAPE NTAPE3, 10                                    HM140798
      DO  120 I=1,NSUR                                                  HM140799
                                                                        HM140800
        REAO INPUT TAPE NTAPE2, 1, NSIZE                                HM140801
        READ INPUT TAPE NTAPE2, 1, (LDW(J),LHIGH(J),J=1,NSIZE)          HM140802
                                                                        HM140803
        DO  119  J=1,NSIZE                                              HM140804
          DO  118  K=1,NSIZE                                            HM140805
  118       A(K,J)=0.                                                   HM140806
      N1=LDW(J)                                                         HM140807
```

```
      N2=LHIGH(J)                                                       HM140808
  119     READ INPUT TAPE NTAPE2, 2, (A(N,J),N=N1,N2)                   HM140809
      WRITE TAPE NTAPE4, NSIZE, NSIZE,  ((A(N,J),N=1,NSIZE),J=1,NSIZE)  HM140810
                                                                        HM140811
        WRITE OUTPUT TAPE NTAPE3, 13, I, NSIZE                          HM140812
        CALL MPRINT (A,NSIZE,NSIZE,MAXR,NTAPE3)                         HM140813
      NSIZES(I)=NSIZE                                                   HM140814
  120   K1=K1+NSIZE                                                     HM140815
                                                                        HM140816
C***********************************************************************HM140817
C NPOINT = TOTAL NUMBER OF CONTROL POINTS ON ALL SURFACES.              HM140818
      NPOINT=K1                                                         HM140819
                                                                        HM140820
C***********************************************************************HM140821
C READ IN RIGID BODY MODAL MATRIX FOR EACH SYSTEM.  (HR)                HM140822
      IF ( NFUS )          121,123,121                                  HM140823
  121 READ INPUT TAPE NTAPE2, 1, NSIZE                                  HM140824
      DO  122  J=1,NRIGID                                               HM140825
  122   READ INPUT TAPE NTAPE2, 2, (HR(I,J),I=1,NSIZE)                  HM140826
      WRITE OUTPUT TAPE NTAPE3, 18, NSIZE                               HM140827
      CALL MPRINT (HR,NSIZE,NRIGID,MAXS,NTAPE3)                         HM140828
      WRITE TAPE NTAPE6, NSIZE, NRIGID, ((HR(I,J),I=1,NSIZE),J=1,NRIGID)HM140829
  123 IF ( NRIGID )          124,128,124                                HM140830
  124 K1=1                                                              HM140831
      WRITE OUTPUT TAPE NTAPE3, 12                                      HM140832
      DO  126  ISUR=1,NSUR                                              HM140833
        READ INPUT TAPE NTAPE2, 1, NSIZE                                HM140834
        K2=K1+NSIZE-1                                                   HM140835
        DO  125  J=1,NRIGID                                             HM140836
  125     READ INPUT TAPE NTAPE2, 2, (HR(I,J),I=K1,K2)                  HM140837
        WRITE OUTPUT TAPE NTAPE3, 13, ISUR,NSIZE                        HM140838
          CALL MPRINT  (HR(K1,1),NSIZE,NRIGID,MAXS,NTAPE3)              HM140839
  126     K1=K1+NSIZE                                                   HM140840
                                                                        HM140841
      IF  ( K1-NPOINT-1)    127,128,127                                 HM140842
  127 WRITE OUTPUT TAPE NTAPE3, 11, K1, NPOINT                          HM140843
                                                                        HM140844
C***********************************************************************HM140845
```

```
C READ IN FLEXIBLE MODAL MATRIX FOR EACH SURFACE, STORE SYSTEM ON NTAPE5HM140846
                                                                       HM140847
  128 K1=0                                                             HM140848
      K2=1                                                             HM140849
      KS=0                                                             HM140850
      KCANT = NFREE+1                                                  HM140851
      REWIND NTAPE4                                                    HM140852
      NSRFLX = NSRFLX                                                  HM140853
      IF ( NFREE )          129,131,129                                HM140854
  129 DO  130  J=1,NSRFLX                                              HM140855
        DO  130  I=1,MAXR                                              HM140856
          H (I,J)=0.                                                   HM140857
  130     FFM(I,J)=0.                                                  HM140858
                                                                       HM140859
  131   WRITE OUTPUT TAPE NTAPE3, 14                                   HM140860
      DO  153  I=1,NSUR                                                HM140861
                                                                       HM140862
        READ INPUT TAPE NTAPE2, 1, NSIZE, NMODES, NFORM                HM140863
                                                                       HM140864
        IF  ( NFREE )       132,139,132                                HM140865
  132 L=NFREE+1                                                        HM140866
      IF  (L-NSRFLX)        1321,1321,134                              HM140867
 1321 DO  133  M=L,NSRFLX                                              HM140868
        DO  133  J=1,NSIZE                                             HM140869
  133     C(J,M)=0.                                                    HM140870
  134 IF  ( I-1 )           136,135,136                                HM140871
  135 CALL MREAD    (FREQ,NSRFLX,1,NFORM,0,0,1,G,NSRFLX,NTAPE2,NTAPE3) HM140872
  136 CALL MREAD (C,NSIZE,NFREE,NFORM,0,1,1,G,MAXR,NTAPE2,NTAPE3)      HM140873
      L=NMODES-NFREE                                                   HM140874
      IF  ( L )             138,138,137                                HM140875
  137 CALL MREAD (C(1,KCANT), NSIZE,L,NFORM,0,1,1,G,MAXR,NTAPE2,NTAPE3) HM140876
      KCANT=KCANT+L                                                    HM140877
  138 NMODES=NSRFLX                                                    HM140878
      K2=1                                                             HM140879
      GOTO  140                                                        HM140880
                                                                       HM140881
  139 CALL MREAD (FREQ(K2),NMODES,1,NFORM,0,0,1,G,NMODES,NTAPE2,NTAPE3) HM140882
                                                                       HM140883
```

```
      CALL MREAD (C,NSIZE,NMODES,NFORM,0,1,1,G,MAXR,NTAPE2,NTAPE3)      HM140884
  140 WRITE OUTPUT TAPE NTAPE3, 13, I,NSIZE                             HM140885
    K=K2+NMODES  -1                                                     HM140886
    WRITE OUTPUT TAPE NTAPE3,  9,  NMODES, (FREQ(J),J=K2,K)             HM140887
      CALL MPRINT  (C,NSIZE,NMODES,MAXR,NTAPE3)                         HM140888
      WRITE TAPE NTAPE5, NSIZE, NMODES, ((C(J,K),J=1,NSIZE),K=1,        HM140889
     1                   NMODES)                                        HM140890
C***********************************************************************HM140891
C READ MASS (M), AND COMPUTE  (M)*(H F)                                  HM140892
    READ TAPE NTAPE4,  J,J,  ((A(K,J),K=1,NSIZE),J=1,NSIZE)             HM140893
                                                                        HM140894
    CALL MMULTD (A,0,C,0,SAVE,NSIZE,NSIZE,NMODES,MAXR,MAXR,MAXR)        HM140895
C***********************************************************************HM140896
C  IF REQUESTED, COMPUTE  (HR)T * (M) * (HF)                            HM140897
    IF  ( NFREE )          141,146,141                                  HM140898
  141 IF  ( NRIGID)  142,146,142                                        HM140899
                                                                        HM140900
  142 L=NFREE+1                                                         HM140901
    DO  143  N=1,NSIZE                                                  HM140902
      DO  143  J=1,NRIGID                                               HM140903
        K=K1+N                                                          HM140904
  143   G(J,N)=HR(K,J)                                                  HM140905
                                                                        HM140906
    CALL MMULTD (G,0,SAVE,0,ARF,NRIGID,NSIZE,NFREE,MAXQ,MAXR,MAXQ)      HM140907
    IF ( NCANT )          146,146,144                                   HM140908
  144 DO  145  N=1,NRIGID                                               HM140909
      DO  145  J=L,NSRFLX                                               HM140910
  145   ARF(N,J)=0.                                                     HM140911
                                                                        HM140912
C***********************************************************************HM140913
C COMPUTE (M FF) =  (H F)TRANSPOSED *  (M) * (H F)                      HM140914
  146 DO  147  M=1,NSIZE                                                HM140915
      DO  147  J=1,NMODES                                               HM140916
  147   A(J,M)=C(M,J)                                                   HM140917
    CALL MMULTD (A,0,SAVE,0,C,NMODES,NSIZE,NMODES,MAXR,MAXR,MAXR)       HM140918
    IF  ( NFREE )          148,150,148                                  HM140919
  148 DO  149  M=1,NSRFLX                                               HM140920
      DO  1481  J=1,NRIGID                                              HM140921
```

```
 1481      HT(J,M)=HT(J,M)+ARF(J,M)                                         HM140922
         DO  149  J=1,NSRFLX                                                HM140923
  149      FFM(M,J)=FFM(M,J)+C(M,J)                                         HM140924
       GOTO  152                                                            HM140925
  150 WRITE OUTPUT TAPE NTAPE3, 22, I                                       HM140926
       CALL MPRINT (C,NMODES,NMODES,MAXR,NTAPE3)                            HM140927
       DO  151  J=1,NMODES                                                  HM140928
         KS=KS+1                                                            HM140929
  151    SPRING(KS)=C(J,J)                                                  HM140930
  152 K1=K1+NSIZE                                                           HM140931
  153   K2=K2+NMODES                                                        HM140932
                                                                            HM140933
       NFLEX=K2-1                                                           HM140934
                                                                            HM140935
       IF ( K1-NPOINT )       158,159,158                                   HM140936
  158 WRITE OUTPUT TAPE NTAPE3, 11, K1, NPOINT                              HM140937
  159 IF  ( NRIGID )         160,1591,160                                   HM140938
 1591 IF  ( NFREE )        1592,164,1592                                    HM140939
 1592 NFLEX=NSRFLX                                                          HM140940
       GOTO  154                                                            HM140941
  160 WRITE TAPE NTAPE4, K1, NRIGID, ((HR(N,J),N=1,K1),J=1,NRIGID)          HM140942
C*******************************************************************HM140943
C COMPUTE  (A RF) = (DELTA M)INVERSE * (H R)TRANSPOSED *(M) * (H F)         HM140944
       IF  ( NFREE )         161,164,161                                    HM140945
  161 DO  162  I=1,NRIGID                                                   HM140946
         DO  162  J=1,NRIGID                                                HM140947
  162      DQ(I,J)=DM(I,J)                                                  HM140948
       NFLEX=NSRFLX                                                         HM140949
       IF  ( NDELM )          1621,163,1621                                 HM140950
 1621   CALL INVERS  (DQ,NRIGID,IR)                                         HM140951
         IF ( IR )           310,163,310                                    HM140952
  163 CALL MMULTD  (DQ,0,HT,0,G,NRIGID,NRIGID,NFLEX,MAXQ,MAXQ,MAXQ)         HM140953
       DO  1630  N=1,NRIGID                                                 HM140954
         DO  1630  J=1,NFLEX                                                HM140955
 1630      G(N,J)=-G(N,J)                                                   HM140956
       WRITE TAPE NTAPE4, NRIGID, NFLEX, ((G(N,J),N=1,NRIGID),J=1,NFLEX) HM140957
                                                                            HM140958
       DO  1631  I=1,NFLEX                                                  HM140959
```

```
      DO  1631  J=1,NRIGID                                               HM140960
 1631    A(I,J)=G(J,I)                                                   HM140961
     CALL MMULTD(A,0,OM,0,C,NSRFLX,NRIGID,NRIGID,MAXR,MAXQ,MAXR)         HM140962
     CALL MMULTD (C,0,G,0,A,NSRFLX,NRIGID,NSRFLX,MAXR,MAXQ,MAXR)         HM140963
     DO  1632  I=1,NSRFLX                                                HM140964
      DO  1632  J=1,NSRFLX                                               HM140965
 1632    FFM(I,J)=FFM(I,J)+A(I,J)                                        HM140966
                                                                         HM140967
  154 DO  155  M=1,NSRFLX                                                HM140968
  155   SPRING(M)=FFM(M,M)                                               HM140969
      WRITE OUTPUT TAPE NTAPE3, 22,                                      HM140970
      CALL MPRINT (FFM,NSRFLX,NSRFLX,MAXR,NTAPE3)                        HM140971
                                                                         HM140972
  164 NFLEX2=NC*NFLEX                                                    HM140973
      IF  ( NAERO )          166,165,166                                 HM140974
  165 NRHO=1                                                             HM140975
      DO  500  I=1,NRIGID                                                HM140976
        DO  500  J=1,NR2                                                 HM140977
  500     DQ(I,J)=0.                                                     HM140978
       NCX=NC                                                            HM140979
      REWIND NTAPE9                                                      HM140980
      GOTO  215                                                          HM140981
C*************************************************************************HM140982
C READ IN WEIGHTING MATRIX FOR EACH SURFACE.....STORE ON NTAPE6,          HM140983
  166   WRITE OUTPUT TAPE NTAPE3, 15                                     HM140984
      DO  178  I=1,NSUR                                                  HM140985
        IF  ( ISW(I) )       167,177,167                                 HM140986
  167   N1=ISXST(I)                                                      HM140987
        IF  ( N1 )           168,172,168                                 HM140988
  168   DO  170  J=1,N1                                                  HM140989
          DO  169  L=1,MAXR                                              HM140990
            A(J,L)=0.                                                    HM140991
  169       A(L,J)=0.                                                    HM140992
  170     A(J,J)=1.                                                      HM140993
                                                                         HM140994
        READ INPUT TAPE NTAPE2, 1, NXST, J, IFORM, IROW                  HM140995
        IF  ( NXST )         171,172,171                                 HM140996
  171   CALL MREAD (A,NXST,NXST,IFORM,IROW,0,1,F,MAXR,NTAPE2,NTAPE3)     HM140997
```

```
                                                                        HM140998
 172   K=N1+1                                                           HM140999
       READ INPUT TAPE NTAPE2, 1, NSIZE, NPART, IFDRM, IROW             HM141000
       IF  ( IFDRM )        174,173,174                                 HM141001
 173   CALL MREAD  (A(K,K),NSIZE,NSIZE,0,0,0,1,F,MAXR,NTAPE2,NTAPE3)    HM141002
       K=K+NSIZE                                                        HM141003
       GDTO  176                                                        HM141004
                                                                        HM141005
 174   DO  175  J=1,NPART                                               HM141006
         READ INPUT TAPE NTAPE2, 1, NSIZE                               HM141007
      N=K+NC                                                            HM141008
      DD  1741  M=K,N                                                   HM141009
       DD  1741  L=1,K                                                  HM141010
1741     A(L,M)=0.                                                      HM141011
       CALL MREAD (A(K,K),NSIZE,NSIZE,1,IRDW,0,0,F,MAXR,NTAPE2,NTAPE3)  HM141012
 175     K=K+NSIZE                                                      HM141013
                                                                        HM141014
 176   K=K-1                                                            HM141015
C***********************************************************************HM141016
       WRITE DUTPUT TAPE NTAPE3, 13, I, K                               HM141017
       CALL MPRINT (A,K,K,MAXR,NTAPE3)                                  HM141018
       WRITE TAPE NTAPE6,  K, K, ((A(J,L),J=1,K),L=1,K)                 HM141019
      GDTD  178                                                         HM141020
 177  WRITE OUTPUT TAPE NTAPE3, 21, I                                   HM141021
 178   CONTINUE                                                         HM141022
 179  REWIND NTAPE6                                                     HM141023
      REWIND NTAPE7                                                     HM141024
      REWIND NTAPE9                                                     HM141025
      NCX=NC                                                            HM141026
C***********************************************************************HM141027
C IF SERIES UF DENSITIES FOR EACH V/8 OMEGA, READ IN THAT SERIES.       HM141028
                                                                        HM141029
      IF  ( NDENS )        180,181,180                                  HM141030
 180  NRHD=NDENS                                                        HM141031
      GDTD  182                                                         HM141032
                                                                        HM141033
 181  READ INPUT TAPE NTAPE2, 1,   NRHO                                 HM141034
      READ INPUT TAPE NTAPE2, 2,   (RHD(I),I=1,NRHD)                    HM141035
```

```
                                                                        HM141036
C***********************************************************************HM141037
C READ IN COMPLEX AERODYNAMIC MATRIX FOR EACH SURFACE                    HM141038
  182 NVEL=NVEL+1                                                       HM141039
      DO  210  I=1,NSURFS                                               HM141040
      K=1                                                               HM141041
      K2=1                                                              HM141042
      IF  ( NFUS )            183,184,183                               HM141043
  183 IF  ( I-1 )             184,188,184                               HM141044
  184 L=I-NFUS                                                          HM141045
      IF  ( ISXST(L) )        185,188,185                               HM141046
  185   K=ISXST(L)+1                                                    HM141047
      K2=2*ISXST(L)+1                                                   HM141048
        DO  186  J=1,K                                                  HM141049
          DO  186  L=1,K2                                               HM141050
  186       A(J,L)=0.                                                   HM141051
                                                                        HM141052
        READ INPUT TAPE NTAPE2, 1, NXST, J, IFORM, IROW                 HM141053
        IF  ( NXST )          187,188,187                               HM141054
                                                                        HM141055
  187   N=  NXST *NC                                                    HM141056
        CALL MREAD (A,NXST,N,IFORM,IROW,0,1,F,MAXR,NTAPE2,NTAPE3)       HM141057
                                                                        HM141058
  188   READ INPUT TAPE NTAPE2, 2, VELC                                 HM141059
        READ INPUT TAPE NTAPE2, 1, NSIZE, NPART, IFORM, IROW            HM141060
        IF  ( IFORM )          190,189,190                              HM141061
  189   N=  NSIZE     *NC                                               HM141062
        CALL MREAD (A(K,K2),NSIZE,N,0,0,0,1,F,MAXR,NTAPE2,NTAPE3)       HM141063
      NSIZE=NSIZE+K-1                                                   HM141064
        GOTO  193                                                       HM141065
                                                                        HM141066
  190   DO  192  J=1,NPART                                              HM141067
          READ INPUT TAPE NTAPE2, 1, NSIZE                              HM141068
      N=K2+NC                                                           HM141069
      DO  191  M=K2,N                                                   HM141070
        DO  191  L=1,K                                                  HM141071
  191     A(L,M)=0.                                                     HM141072
          N=  NSIZE      *NC                                            HM141073
```

```
      CALL MREAD (A(K,K2),NSIZE,N,1,IROW,0,0,F,MAXR,NTAPE2,NTAPE3)      HM141074
        K=K+NSIZE                                                       HM141075
  192   K2=K2+N                                                         HM141076
      NSIZE=K-1                                                         HM141077
        N=K2-1                                                          HM141078
  193 IF  ( I-1 )            199,194,199                                HM141079
  194 IF  ( NFUS )           195,197,195                                HM141080
  195  WRITE OUTPUT TAPE NTAPE3, 16, NSIZE                              HM141081
      CALL MPRINT (A,NSIZE,N,MAXR,NTAPE3)                               HM141082
                                                                        HM141083
C***********************************************************************HM141084
C  COMPUTE (DQ) = (HR)T * (CH) * (HR)                                   HM141085
                                                                        HM141086
      K1=1                                                              HM141087
      K2=1                                                              HM141088
      M=NC-1                                                            HM141089
      READ TAPE NTAPE6, NSIZE, L, ((H(L,J),L=1,NSIZE),J=1,NRIGID)       HM141090
                                                                        HM141091
      CALL MMULTD (A,M,H,O,F,NSIZE,NSIZE,NRIGID,MAXR,MAXR,MAXR)         HM141092
                                                                        HM141093
      DO  196  L=1,NSIZE                                                HM141094
        DO  196  J=1,NRIGID                                             HM141095
  196     A(J,L)=H(L,J)                                                 HM141096
                                                                        HM141097
      CALL MMULTD (A,O,F,M,DQ,NRIGID,NSIZE,NRIGID,MAXR,MAXR,MAXQ)       HM141098
                                                                        HM141099
      GOTO  210                                                         HM141100
                                                                        HM141101
  197 DO  198  L=1,NRIGID                                               HM141102
        DO  198  J=1,NR2                                                HM141103
  198     DQ(L,J)=0.                                                    HM141104
                                                                        HM141105
                                                                        HM141106
  199 L=I-NFUS                                                          HM141107
      IF  ( ISW(L) )         202,200,202                                HM141108
  200 WRITE TAPE NTAPE7,  NSIZE, N,    ((A(J,M),J=1,NSIZE),M=1,N)       HM141109
      GOTO  208                                                         HM141110
                                                                        HM141111
```

```
202 READ TAPE NTAPE6, L, L,  ((T(J,M),J=1,L),M=1,L)

    IF ( NSIZE-L )         204,206,204
204 WRITE OUTPUT TAPE NTAPE3, 11, NSIZE,L

206 CALL MMULTD (T,O,A,NC-1,F,L,L,L,MAXR,MAXR,MAXR)
    WRITE TAPE NTAPE7,  NSIZE, N,    ((F(J,M),J=1,NSIZE),M=1,N)
208 WRITE OUTPUT TAPE NTAPE3, 17, VELC
    L=I-NFUS
    WRITE OUTPUT TAPE NTAPE3, 13,  L, NSIZE
    CALL MPRINT (A,NSIZE,N,MAXR,NTAPE3)

210 CONTINUE

C***********************************************************************
C CARRY ON FROM HERE TO END ONCE FOR EACH DENSITY.

215 DO  300  IRHO=1,NRHO
    IF  ( NFREE )         216,219,216
216 DO  217  I=1,NFLEX
      DO  217  J=1,NFLEX2
217     SAVE(I,J)=0.
219 REWIND NTAPE4
    REWIND NTAPE5
    REWIND NTAPE7
    REWIND NTAPE8
      CON=RHO(IRHO)*BR(1)**2*S(1)  *32.174
    IF  ( NRIGID)   218,222,218
218   DO  220  I=1,NRIGID
        DO  220  J=1,NR2,NCX
          DMBAR(I,J+1)=CON*DQ(I,J+1)
          K=J/2+1
          DMBAR(I,J)=OM(I,K)+CON*DQ(I,J)
          BARMRR(I,J)=DMBARII,J)
220       BARMRR(I,J+1)=DMBAR(I,J+1)
222   DO  236  ISUR=1,NSUR
        K=ISUR+NFUS
```

```
      CON=RHO(IRHO)*BR(K)**2*S(K) *32.174                               HM141150
                                                                        HM141151
C***********************************************************************HM141152
C              READ  (M) I                                              HM141153
                                                                        HM141154
      READ TAPE NTAPE4, NSIZE, NSIZE, ((F(I,J),I=1,NSIZE),J=1,NSIZE)HM141155
      IF ( NAERO )           221,221,223                                HM141156
  221 DO  226  I=1,NSIZE                                                HM141157
        DO  226  J=1,NSIZE                                              HM141158
  226   A(I,J)=F(I,J)                                                   HM141159
      L=NSIZE                                                           HM141160
      GOTO  231                                                         HM141161
                                                                        HM141162
C***********************************************************************HM141163
C              READ  (W)*(CH) I                                         HM141164
                                                                        HM141165
  223   READ TAPE NTAPE7, NSIZE, L,  ((A(I,J),I=1,NSIZE),J=1,L)         HM141166
                                                                        HM141167
C***********************************************************************HM141168
C         COMPUTE (M BAR) =((M)+RHO*BR**2*S*(W)*(CH) ) I                HM141169
                                                                        HM141170
      DO  228  I=1,NSIZE                                                HM141171
        DO  228  J=1,L,NCX                                              HM141172
        K=J/2+1                                                         HM141173
        A(I,J)=CON*A(I,J)+F(I,K)                                        HM141174
  228   A(I,J+1)=CON*A(I,J+1)                                           HM141175
  231 IF  ( NRIGID )           229,233,229                              HM141176
  229   WRITE TAPE NTAPE8, NSIZE, L, ((A(I,J),I=1,NSIZE),J=1,L)         HM141177
                                                                        HM141178
C***********************************************************************HM141179
C READ  (H F)  INTO CORE STORAGE                                        HM141180
                                                                        HM141181
  233   READ TAPE NTAPE5, NSIZE, NMODES, ((T(I,J),I=1,NSIZE),           HM141182
     1                        J=1,NMODES)                               HM141183
                                                                        HM141184
C***********************************************************************HM141185
C COMPUTE   (M BAR) * (H F)                                             HM141186
      N=NC-1                                                            HM141187
```

```
      M=  NMODES   *NC                                                   HM141188
    IF ( NRIGID )          224,225,224                                   HM141189
                                                                         HM141190
 224    CALL MMULTD (A,N,T,O,F,NSIZE,NSIZE,NMODES,MAXR,MAXR,MAXR)        HM141191
        WRITE TAPE NTAPE8, NSIZE, M, ((F(I,J),I=1,NSIZE),J=1,M)          HM141192
 225    DO  232  I=1,NSIZE                                               HM141193
          DO  232  J=1,NMODES                                            HM141194
 232        F(J,I)=T(I,J)                                                HM141195
                                                                         HM141196
C***********************************************************************HM141197
C COMPUTE   (H F)T * (M BAR)                                             HM141198
        CALL MMULTD (F,O,A,N,T,NMODES,NSIZE,NSIZE,MAXR,MAXR,MAXR)        HM141199
    IF ( NRIGID )          227,230,227                                   HM141200
 227    WRITE TAPE NTAPE8, NMODES, L, ((T(I,J),I=1,NMODES),J=1,L)        HM141201
 230    DO  234  I=1,NSIZE                                               HM141202
          DO  234  J=1,NMODES                                            HM141203
 234        A(I,J)=F(J,I)                                                HM141204
                                                                         HM141205
C***********************************************************************HM141206
C     THEN   (H F)T * (M BAR) * (H F)                                    HM141207
          CALL MMULTD (T,N,A,O,F,NMODES,NSIZE,NMODES,MAXR,MAXR,MAXR)     HM141208
    IF  ( NFREE )          428,235,428                                   HM141209
 428 DO  430  I=1,NFLEX                                                  HM141210
       DO  430  J=1,NFLEX2                                               HM141211
 430    SAVE(I,J)=SAVE(I,J)+F(I,J)                                       HM141212
    GOTO  236                                                            HM141213
 235      WRITE TAPE NTAPE9, NMODES, M, ((F(I,J),I=1,NMODES),J=1,M)      HM141214
 236      CONTINUE                                                       HM141215
                                                                         HM141216
    IF  ( NFREE )          432,237,432                                   HM141217
 432 WRITE TAPE NTAPE9, NFLEX, NFLEX2, ((SAVE(I,J),I=1,NFLEX),           HM141218
    1                    J=1,NFLEX2)                                     HM141219
                                                                         HM141220
C***********************************************************************HM141221
C  IF NRIGID=0, THATS ALL FOR THIS DENSITY.                              HM141222
C            IF NOT, READ HR MATRIX FROM NTAPE4                          HM141223
                                                                         HM141224
 237   REWIND NTAPE8                                                     HM141225
```

```
      IF  ( NRIGID)  238,300,238                                        HM141226
  238 READ TAPE NTAPE4,  NPOINT, NRIGID, ((HR(I,J),I=1,NPOINT),         HM141227
     1                   J=1,NRIGID)                                    HM141228
      K1=0                                                              HM141229
      K2=1                                                              HM141230
      DO  244  ISUR=1,NSUR                                              HM141231
                                                                        HM141232
C***********************************************************************HM141233
C  READ (M BAR) AND COMPUTE  (H R)TRANSPOSED * (M BAR) * (H R) +(DELTAM)HM141234
        READ TAPE NTAPE8, NSIZE, M,  ((A(I,J),I=1,NSIZE),J=1,M)         HM141235
        DO  240  I=1,NSIZE                                              HM141236
          DO  240  J=1,NRIGID                                           HM141237
            K=K1+I                                                      HM141238
  240       G(J,I)=HR(K,J)                                              HM141239
        CALL MMULTD (G,O,A,N,HT,NRIGID,NSIZE,NSIZE,MAXQ,MAXR,MAXQ)      HM141240
        CALL MMULTD (HT,N,HR(K1+1,1),O,A,NRIGID,NSIZE,NRIGID,MAXQ,      HM141241
     1               MAXS,MAXR)                                         HM141242
        DO  242  I=1,NRIGID                                             HM141243
          DO  242  J=1,NR2                                              HM141244
  242       BARMRR(I,J)=BARMRR(I,J)+A(I,J)                              HM141245
                                                                        HM141246
C***********************************************************************HM141247
C       FIND    (H R)TRANSPOSED * (M BAR) * (H F)                       HM141248
C               (H F)TRANSPOSED * (M BAR) * (H R)   AND STORE ON NTAPE9 HM141249
        READ TAPE NTAPE8, NSIZE, M,  ((A(I,J),I=1,NSIZE),J=1,M)         HM141250
        CALL MMULTD  (G,O,A,N,HT,NRIGID,NSIZE,NFLEX,MAXQ,MAXR,MAXQ)     HM141251
                                                                        HM141252
        WRITE TAPE NTAPE9, NRIGID,M,((HT(I,J),I=1,NRIGID),J=1,M)        HM141253
        READ TAPE NTAPE8, NMODES, L, ((A(I,J),I=1,NMODES),J=1,L)        HM141254
        CALL MMULTD (A,N,HR(K1+1,1),O,T,NMODES,NSIZE,NRIGID,MAXR,MAXS,HM141255
     1              MAXR)                                               HM141256
        WRITE TAPE NTAPE9, NMODES, NR2, ((T(I,J),I=1,NMODES),J=1,NR2)   HM141257
  244   K1=K1+NSIZE                                                     HM141258
                                                                        HM141259
      IF  ( NFREE )         246,250,246                                 HM141260
  246 READ TAPE NTAPE4, NSIZE, NMODES, ((ARF(I,J),I=1,NSIZE),J=1,NMODES)HM141261
      CALL MMULTD (DMBAR,N,ARF,O,HT,NRIGID,NRIGID,NMODES,MAXQ,MAXQ,MAXQ)HM141262
      DO  248  I=1,NRIGID                                               HM141263
```

```
      DO  248  J=1,NMODES                                                HM141264
  248   SAVE(J,I) = ARF(I,J)                                             HM141265
      CALL MMULTD (SAVE,0,HT,N,A,NMODES,NRIGID,NMODES,MAXR,MAXQ,MAXR)   HM141266
      M=  NMODES  *NC                                                    HM141267
      WRITE TAPE NTAPE9, NMODES, M, ((A(I,J),I=1,NMODES),J=1,M)         HM141268
      WRITE TAPE NTAPE9, NRIGID, M, ((HT(I,J),I=1,NRIGID),J=1,M)        HM141269
      CALL MMULTD (SAVE,0,DMBAR,N,A,NMODES,NRIGID,NRIGID,MAXR,MAXQ,MAXR)HM141270
      WRITE TAPE NTAPE9, NMODES, NR2, ((A(I,J),I=1,NMODES),J=1,NR2)     HM141271
                                                                         HM141272
C***********************************************************************HM141273
C COMPUTE  (M BAR RR) INVERSE, AND STORE ON TAPE NTAPE9.                 HM141274
  250 GOTO (256,252),NCX                                                 HM141275
  252 DO  254  I=1,NRIGID                                                HM141276
        DO  254  J=1,NR2,NCX                                             HM141277
          K=J/2+1                                                        HM141278
          G(I,K)=BARMRR(I,J+1)                                           HM141279
  254     BARMRR(I,K)=BARMRR(I,J)                                        HM141280
                                                                         HM141281
  256 CALL MNVRSX (BARMRR,G,ARF(1,1),ARF(1,MAXQ),NRIGID,IR,NC-1)         HM141282
                                                                         HM141283
      IF ( IR )  310,258,310                                             HM141284
  258 GOTO (264,260),NCX                                                 HM141285
  260 DO  262  I=1,NRIGID                                                HM141286
        DO  262  J=1,NR2,NCX                                             HM141287
          K=NR2-J                                                        HM141288
          L=K/2+1                                                        HM141289
          BARMRR(I,K)=BARMRR(I,L)                                        HM141290
  262     BARMRR(I,K+1)=G(I,L)                                           HM141291
                                                                         HM141292
  264 WRITE TAPE NTAPE9, NRIGID, NR2, ((BARMRR(I,J),I=1,NRIGID),J=1,NR2)HM141293
                                                                         HM141294
      K=1                                                                HM141295
      DO  266  ISUR=1,NSUR                                               HM141296
      L=NSIZES(ISUR)                                                     HM141297
        CALL MMULTD (HR(K,1),0,BARMRR,NC-1,A,     L     ,NRIGID,NRIGID, HM141298
     1                 MAXS,MAXQ,MAXR)                                   HM141299
      K=K+L                                                              HM141300
  266   WRITE TAPE NTAPE9, L, NR2,  ((A(I,J),I=1,L),J=1,NR2)            HM141301
```

```
                                                                HM141302
  300 CONTINUE                                                  HM141303
                                                                HM141304
      CALL CHAIN (2,NTAPE1)                                     HM141305
  310 WRITE OUTPUT TAPE NTAPE3, 20                              HM141306
      GOTO  100                                                 HM141307
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
4746	11212	11041	25441

STORAGE LOCATIONS FOR VARIABLES APPEARING IN COMMON STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
AITKEN	32044	76454	ARF	21041	51061	A	32041	76451	BARMRR	32187	76673
BOOLT	32473	77331	BREF	32411	77233	BR	32409	77231	C	29541	71545
DMBAR	32259	77003	DM	32367	77157	DQ	32331	77113	EPDP	32047	76457
EPSP	32048	76460	FFM	13541	32345	FREQ	32451	77303	F	27041	64641
G	20201	47351	HR	27041	64641	H	20801	50501	HT	20801	50501
ISW	32541	77435	ISXST	32561	77461	IT	32561	77461	LHIGH	20201	47351
LOW	21041	51061	MAXP	32480	77340	MAXQ	32482	77342	MAXR	32483	77343
MAXS	32481	77341	MODES	32486	77346	NAERO	32115	76563	NCANT	32489	77351
NCARDS	32049	76461	NC	32476	77334	NDENS	32487	77347	NFLEX2	32477	77335
NFLEX	32484	77344	NFREE	32490	77352	NFUS	32488	77350	NGO	32475	77333
NITRDP	32045	76455	NITRSP	32046	76456	NPOINT	32485	77345	NPUNCH	32474	77332
NR2	32478	77336	NRHO	32114	76562	NRIGID	32491	77353	NSIZES	32521	77411
NSRFLX	32410	77232	NSURFS	32479	77337	NSUR	32492	77354	NTAPE0	32050	76462
NTAPE1	32501	77365	NTAPE2	32500	77364	NTAPE3	32499	77363	NTAPE4	32498	77362
NTAPE5	32497	77361	NTAPE6	32496	77360	NTAPE7	32495	77357	NTAPE8	32494	77356
NTAPE9	32493	77355	NVEL	32051	76463	PICON	32113	76561	RHO	32471	77327
SAVE	16041	37251	SPRING	32112	76560	S	32388	77204	T	21041	51061
VELCTY	32072	76510									

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
CON	4745	11211	IFORM	4744	11210	IRHO	4743	11207	IROW	4742	11206
IR	4741	11205	I	4740	11204	ISUR	4739	11203	J	4738	11202
K1	4737	11201	K2	4736	11200	KCANT	4735	11177	K	4734	11176
KS	4733	11175	L	4732	11174	M	4731	11173	N1	4730	11172
N2	4729	11171	NCON	4728	11170	NCX	4727	11167	NDELM	4726	11166
NFORM	4725	11165	NMODES	4724	11164	NPART	4723	11163	N	4722	11162
NSIZE	4721	11161	NXST	4720	11160	VELC	4719	11157			

SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

	EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC
8)1	1	11100	8)2	2	11076	8)3	3	11074	8)4	4	11012
8)5	5	10765	8)6	6	10744	8)7	7	10735	8)8	8	10716
8)9	9	10706	8)A	10	10674	8)8	11	10666	8)C	12	10630
8)D	13	10621	8)E	14	10607	8)F	15	10600	8)G	16	10571
8)H	17	10553	8)I	18	10532	8)J	19	10512	8)K	20	10501
8)L	21	10467	8)M	22	10451	8)N	23	10441	8)O	24	10436

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
1)	4673	11101	2)	4325	10345	3)	4342	10366	4)	32767	77777
6)	4351	10377	A)108	4229	10205	A)108	4238	10216	A)10K	4251	10233
A)10M	4264	10250	A)100	4277	10265	A)10P	4290	10302	A)10S	4303	10317
A)10T	4312	10330	C)G0	4677	11105	C)G2	4678	11106	C)G3	4679	11107
C)G4	4680	11110	C)G6	4681	11111	C)G7	4682	11112	C)G8	4683	11113
C)G9	4684	11114	C)GA	4685	11115	C)100	4686	11116	C)101	4687	11117
C)105	4688	11120	C)106	4689	11121	C)108	4690	11122	C)10A	4691	11123
C)108	4692	11124	C)10K	4693	11125	C)10M	4694	11126	C)10O	4695	11127
C)10P	4696	11130	C)10S	4697	11131	C)10T	4698	11132	C)10U	4699	11133
C)201	4700	11134	C)202	4701	11135	C)203	4702	11136	C)204	4703	11137
C)205	4704	11140	C)206	4705	11141	C)207	4706	11142	C)208	4707	11143
C)209	4708	11144	C)20A	4709	11145	C)20B	4710	11146	C)20C	4711	11147
C)20D	4712	11150	C)20E	4713	11151	C)20F	4714	11152	C)20G	4715	11153
C)20H	4716	11154	C)20I	4717	11155	C)20J	4718	11156	D)100	356	00544
D)12M	953	01671	D)16I	2163	04163	D)190	2984	05650	D)1AM	3360	06440
D)1CK	4030	07676	D)25A	1683	03223	D)263	1997	03715	D)26R	2329	04431
D)2AR	3402	06512	D)2C9	3890	07462	D)2CH	3996	07634	D)2CI	4000	07640
D)2CD	4084	07764	D)2CT	4133	10045	D)36I	2162	04162	D)36R	2328	04430
D)390	2983	05647	D)3AR	3401	06511	D)3CD	4083	07763	D)3CT	4132	10044
D)403	98	00142	D)418	484	00744	D)436	1174	02226	D)43F	1271	02367
D)45I	1752	03330	D)45L	1821	03435	D)461	1992	03710	D)46L	2225	04261
D)47T	2634	05112	D)47V	2642	05122	D)488	2720	05240	D)48C	2775	05327
D)48I	2841	05431	D)48S	2930	05562	D)4CG	3992	07630	D)4D5	4221	10175
D)56L	2224	04260	D)588	2719	05237	D)5D5	4220	10174	D)636	1173	02225
D)67V	2641	05121	D)76I	2161	04161	D)790	2982	05646	E)25	776	01410

E)2J	907	01613	E)2L	928	01640	E)2M	952	01670	E)2O	991	01737
E)34	1155	02203	E)3R	1351	02507	E)5B	1686	03226	E)7C	2477	04655
E)8N	2865	05461	E)96	3022	05716	E)9G	3115	06053	E)9Q	3185	06161
E)A4	3244	06254									

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
CHAIN	17	00021	INVERS	15	00017	MMULTO	14	00016	MNVRSX	16	00020
MPRINT	8	00010	MREAD	11	00013	ROLN	3	00003	SETH	1	00001
(FIL)	7	00007	(FPT)	0	00000	(RLR)	13	00015	(RTN)	5	00005
(RWT)	2	00002	(STB)	9	00011	(STH)	6	00006	(TSB)	12	00014
(TSH)	4	00004	(WLR)	10	00012						

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

CHAIN	INVERS	MMULTO	MNVRSX	MPRINT	MREAD	RDLN	SETH
(FIL)	(FPT)	(RLR)	(RTN)	(RWT)	(STB)	(STH)	(TSH)
(TSH)	(WLR)						

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
98	31	00034	99	33	00042	100	61	00143	1011	68	00241
1012	72	00305	102	76	00321	103	79	00334	104	80	00342
105	81	00344	106	100	00437	107	105	00456	108	106	00460
109	108	00474	110	112	00520	111	115	00545	112	116	00547
113	117	00553	114	119	00566	115	121	00576	116	122	00614
117	131	00653	118	143	00745	119	146	00756	120	165	01074
121	168	01111	122	171	01135	123	189	01230	124	190	01232
125	197	01314	126	206	01364	127	208	01377	128	210	01411
129	217	01451	130	220	01471	131	221	01502	132	226	01534
1321	228	01545	133	230	01601	134	231	01614	135	232	01621
136	234	01641	137	238	01672	138	241	01724	139	244	01740
140	248	01775	141	280	02204	142	281	02206	143	285	02256
144	289	02323	145	291	02360	146	292	02370	147	294	02406
148	298	02440	1481	300	02465	149	302	02473	150	304	02510

151	310	02542	152	311	02546	153	312	02553	158	315	02575
159	317	02605	1591	318	02611	1592	319	02613	160	321	02616
161	331	02656	162	333	02671	1621	336	02704	163	339	02714
1630	343	02747	1631	355	03032	1632	362	03113	154	363	03123
155	364	03127	164	369	03146	165	371	03156	500	374	03173
166	378	03211	167	381	03227	168	383	03233	169	386	03254
170	387	03263	171	391	03311	172	393	03331	173	397	03372
174	401	03431	1741	407	03507	175	410	03541	176	411	03561
177	426	03675	178	428	03711	179	429	03716	180	434	03732
181	436	03735	182	443	03766	183	448	04026	184	449	04031
185	451	04040	186	455	04103	187	459	04136	188	462	04164
189	467	04222	190	472	04254	191	478	04333	192	483	04406
193	486	04432	194	487	04437	195	488	04441	196	508	04570
197	512	04623	198	514	04636	199	515	04645	200	517	04656
202	527	04715	204	537	04761	206	539	04771	208	550	05054
210	557	05113	215	558	05123	216	560	05131	217	562	05144
219	563	05153	218	569	05177	220	575	05304	222	576	05330
221	589	05432	226	591	05445	223	594	05462	228	607	05623
231	608	05651	229	609	05653	233	618	05717	224	630	05772
225	641	06054	232	643	06072	227	647	06122	230	656	06162
234	658	06200	428	662	06230	430	664	06243	235	666	06255
236	675	06313	432	677	06320	237	686	06356	238	688	06362
240	712	06542	242	719	06645	244	760	07127	246	762	07142
248	775	07240	250	808	07463	252	809	07466	254	813	07563
256	814	07605	258	817	07641	260	818	07644	262	823	07742
264	824	07765	266	839	10121	300	848	10165	310	851	10176

```
      DIMENSION  ISXST(20), ISW(20), BR(21), S(21), RHO(20), DM(6,6),   HM141627
     1           DQ(6,12), OMBAR(6,12), BARMRR(6,12), FREQ(40)          HM141628
     2           , NSIZES(20),  IT(520),SPRING(40),VELCTY(20)  ,        HM141629
     3           HFF(40,80), HRF(6,80), HFR(40,12), ARF(40,80),         HM141630
     4           HRR(6,12), HRRR (6,80), A(40,160), GUESS(40,2),        HM141631
     5           VECTOR(40,50), EIGVAL(50), NITER(25), US(40,50),       HM141632
     6           H(40,58), NAKSR(25), NAKDR(25), BOOLT(2)               HM141633
     7           , HF1(49,100), HF(49,40), H1(49,46), OMEGA(25)         HM141634
     8           ,  B(40,6)                                             HM141635
                                                                        HM141636
      EQUIVALENCE  (IT(1),ISXST(1)), (IT(21),ISW(1)), (IT(41),NSIZES)   HM141637
     1            ,(IT(61),NTAPE1), (IT(62),NTAPE2), (IT(63),NTAPE3),   HM141638
     2             (IT(64),NTAPE4), (IT(65),NTAPE5), (IT(66),NTAPE6),   HM141639
     3             (IT(67),NTAPE7), (IT(68),NTAPE8), (IT(69),NTAPE9),   HM141640
     4             (IT(70),NSUR), (IT(71),NRIGID), (IT(72),NFREE),      HM141641
     5             (IT(73),NCANT), (IT(74),NFUS), (IT(75),NDENS),       HM141642
     6             (IT(76),MODE ), (IT(77),NPOINT), (IT(78),NFLEX),     HM141643
     7             (IT(79),MAXR), (IT(80),MAXQ), (IT(81),MAXS),         HM141644
     8             (IT(82),MAXP), (IT(83),NSURFS), (IT(84),NR2),        HM141645
     9             (IT(85),NFLEX2), (IT(86),NC), (IT(87),NGO)           HM141646
      EQUIVALENCE  (IT(88),NPUNCH), (IT(89),BOOLT), (IT(91),RHO),       HM141647
     1             (IT(111),FREQ), (IT(151),BREF), (IT(152),NSRFLX),    HM141648
     2             (IT(153),BR), (IT(174),S), (IT(195),DM), (IT(231),   HM141649
     3             DQ), (IT(303), DMBAR), (IT(375),BARMRR) ,            HM141650
     4             (IT(447),NAERO), (IT(448),NRHO), (IT(449),PICON),    HM141651
     5            (IT(450),SPRING), (IT(490),VELCTY),(IT(511),NVEL),    HM141652
     6           (IT(512),NTAPEO), (IT(513),NCARDS), (IT(514),EPSP),    HM141653
     7           (IT(515),EPDP), (IT(516),NITRSP), (IT(517),NITROP),    HM141654
     8           (IT(518),AITKEN), (A,HFF,HF1), (US,HFR), (H1,H),       HM141655
     9           (A(4901),HF,ARF), (VECTOR,HRRR),(VECTOR(961),HRR)      HM141656
      EQUIVALENCE  (NAKSR,OMEGA), (B,HRF)                               HM141657
                                                                        HM141658
      COMMON     IT, A, VECTOR, HRF, GUESS, EIGVAL, US, NITER, H, NAKSR,HM141659
     1           NAKDR                                                  HM141660
  200 FORMAT  (1H1, 20X, 5H MODE  7X, 12H OMEGA (CPS) 10X,  8H DAMPING  HM141661
     1       4X, 17H VELOCITY (KNOTS)  /// )                            HM141662
  201 FORMAT  (1H   21X,  1I4,  3E20.8)                                 HM141663
```

```
202 FORMAT  (1H1  48X, 12H OUTPUT OATA   // 1H  16X, 7HFLUTTER         HM141664
   1         55H ANALYSIS BY A MOOAL METHOD USING AEROOYNAMIC INFLUENCEHM141665
   2         13H COEFFICIENTS  // 1HO  14X, 11H DENSITY =   1E20.8,     HM141666
   3         5X, 20H REDUCED VELOCITY =   1E20.8,  //  1HO  12X,  1I6,  HM141667
   4        32H RIGIO BOOY OEGREES OF FREEDOM,    12X, 1I6, 6H INPUT    HM141668
   5        16H MCOES (TOTAL).    ////  )                               HM141669
203 FORMAT (1HO 40X, 21H FINAL FLUTTER MOOES    /////  )                HM141670
204 FORMAT (1HO 38X, 8HSURFACE  1I2, 1H,  1I6,  15H CONTROL POINTS  ) HM141671
205 FORMAT  ( 2I4, 62X, 1A6, 1I4 )                                      HM141672
206 FORMAT  ( 1HO  33X, 23H PUNCHEO CARDS NUMBERS  1A6, 1I4,  6H THRU HM141673
   1            1A6, 1I4  )                                             HM141674
207 FORMAT   ( 1HO  45X, 16H DYNAMIC MATRIX  )                          HM141675
208 FORMAT  (5H MOOE   1I6, 32H, GIVES AN IMAGINARY FREQUENCY.    )     HM141676
209 FORMAT   (1HO  /// 1HO  40X, 20H GENERALIZEO MASSES  ////           HM141677
   1           (1H  30X,  5H MASS  1I4,  3H =   1E16.8 )   )            HM141678
                                                                        HM141679
8      BCOZ = 606030440104                                              HM141680
       MAXR = 49                                                        HM141681
                                                                        HM141682
       REWIND NTAPE7                                                    HM141683
       REWIND NTAPE9                                                    HM141684
                                                                        HM141685
       OO  450  IRHO = 1,NRHO                                           HM141686
       REWIND NTAPE5                                                    HM141687
       NVEL=NVEL                                                        HM141688
       MODES=MOOE                                                       HM141689
                                                                        HM141690
       OO  301  I=1,NRIGID                                              HM141691
         OO  301  J=1,NFLEX2                                            HM141692
 301       HRF(I,J)=0.                                                  HM141693
       DO  304  I=1,NFLEX                                               HM141694
         DO  302  J=1,NFLEX2                                            HM141695
 302       HFF(I,J)=0.                                                  HM141696
         DC  304  J=1,NR2                                               HM141697
 304     HFR(I,J)=0.                                                    HM141698
                                                                        HM141699
       L=0                                                              HM141700
       K=C                                                              HM141701
```

```
                                                                        HM141702
      IF  ( NFREE )          290,294,290                                HM141703
  290 READ TAPE NTAPE9, NSIZE, N,  ((HFF(I,J),I=1,NSIZE ),J=1,N)        HM141704
      GOTO  296                                                         HM141705
C***********************************************************************HM141706
C    READ AND POSITION ALL ( M BAR FF )                                 HM141707
  294 DO  307  ISUR=1,NSUR                                              HM141708
        READ TAPE NTAPE9, NSIZE, N, ((ARF(I,J),I=1,NSIZE),J=1,N)        HM141709
        DO  306  I=1,NSIZE                                              HM141710
          K=K+1                                                         HM141711
          DO  305  J=1,N                                                HM141712
            L=L+1                                                       HM141713
  305     HFF(K,L)=ARF(I,J)                                             HM141714
  306     L=L-N                                                         HM141715
  307   L=L+N                                                           HM141716
      IF  ( NRIGID )          296,297,296                               HM141717
  297 DO  298  I=1,NFLEX                                                HM141718
        DO  298  J=1,NFLEX2                                             HM141719
  298     ARF(I,J)=0.                                                   HM141720
      GOTO  350                                                         HM141721
  296 K=0                                                               HM141722
      L=0                                                               HM141723
C***********************************************************************HM141724
C    READ AND POSITION ALL ( M BAR FR )  AND  ( M BAR RF )              HM141725
      DO  312  ISUR=1,NSUR                                              HM141726
                                                                        HM141727
        READ TAPE NTAPE9, NSIZE, N, ((ARF(I,J),I=1,NSIZE ),J=1,N)       HM141728
        DO  309  I=1,NSIZE                                              HM141729
          DO  308  J=1,N                                                HM141730
            L=L+1                                                       HM141731
  308       HRF(I,L)=HRF(I,L)+ARF(I,J)                                  HM141732
  309     L=L-N                                                         HM141733
        L=L+N                                                           HM141734
        READ TAPE NTAPE9, NSIZE, N, ((ARF(I,J),I=1,NSIZE),J=1,N)        HM141735
        DO  310  I=1,NSIZE                                              HM141736
            K=K+1                                                       HM141737
          DO  310  J=1,N                                                HM141738
  310       HFR(K,J)=HFR(K,J)+ARF(I,J)                                  HM141739
```

```
      IF  ( NFREE )          311,312,311                                     HM141740
  311 L=0                                                                    HM141741
      K=D                                                                    HM141742
  312   CCNTINUE                                                             HM141743
                                                                             HM141744
C NFLEX = TDTAL NUMBER DF FLEXIBLE MDDES.                                     HM141745
                                                                             HM141746
      IF  ( NFREE )          314,322,314                                     HM141747
  314 READ TAPE NTAPE9, NSIZE, N, ((ARF(I,J),I=1,NSIZE),J=1,N)               HM141748
      DD  316  I=1,NFLEX                                                     HM141749
        DD  316  J=1,NFLEX2                                                  HM141750
  316     HFF(I,J)=HFF(I,J)+ARF(I,J)                                         HM141751
      READ TAPE NTAPE9, NSIZE, N, ((ARF(I,J),I=1,NSIZE),J=1,N)               HM141752
      DD  318  I=1,NRIGID                                                    HM141753
        DO  318  J=1,NFLEX2                                                  HM141754
  318     HRF(I,J)=HRF(I,J)+ARF(I,J)                                         HM141755
      READ TAPE NTAPE9, NSIZE, N, ((ARF(I,J),I=1,NSIZE),J=1,N)               HM141756
      DD  320  I=1,NFLEX                                                     HM141757
        DD  32D  J=1,NR2                                                     HM141758
  320     HFR(I,J)=HFR(I,J)+ARF(I,J)                                         HM141759
                                                                             HM141760
  322 READ TAPE NTAPE9, NSIZE, N,  ((HRR(I,J),I=1,NSIZE),J=1,N)              HM141761
C*****************************************************************************HM141762
C CDMPUTE  (M BAR FR) * (M BAR RR)INV * (M BAR RF)                           HM141763
      CALL DPMLTX (HRR,NC  ,HRF,NC  ,HRRR ,NRIGID,NRIGID,NFLEX,MAXQ,         HM141764
     1             MAXQ,MAXQ,1)                                              HM141765
      IF  ( NAERO )          326,324,326                                     HM141766
  324 WRITE TAPE NTAPE7,  ((HRRR (I,J),I=1,NRIGID),J=1,NFLEX)                HM141767
  326 CONTINUE                                                               HM141768
      CALL DPMLTX (HFR,NC  ,HRRR ,NC  ,ARF,NFLEX,NRIGID,NFLEX,MAXP,          HM141769
     1             MAXQ,MAXP,1)                                              HM14177D
  35D IF  ( NAERO )          352,351,352                                     HM141771
  351 WRITE TAPE NTAPE7,  ((HFF(I,J),I=1,NFLEX),J=1,NFLEX)                   HM141772
  352 DO  354  I=1,NFLEX                                                     HM141773
        FFM=SPRING(I)*FREQ(I)**2                                             HM141774
        DD  354  J=1,NFLEX2                                                  HM141775
  354     HFF(I,J)= (HFF(I,J)-ARF(I,J)) / FFM                                HM141776
                                                                             HM141777
```

```
C*********************************************************************HM141778
      WRITE OUTPUT TAPE NTAPE3, 202, RHO(IRHO), VELCTY(NVEL), NRIGID,   HM141779
     1                            NFLEX                                 HM141780
      DO  355  I=1,MODES                                                HM141781
        NAKSR(I)=0                                                      HM141782
  355   NAKDR(I)=0                                                      HM141783
      IF  ( NPUNCH )         356,358,358                                HM141784
  356 WRITE OUTPUT TAPE NTAPE3, 207                                     HM141785
      CALL MPRINT (HFF,NFLEX,NFLEX2,MAXP,NTAPE3)                        HM141786
                                                                        HM141787
C*********************************************************************HM141788
  358 CALL MITERS  (HFF,NTAPE8,NFLEX,GUESS,0,MODES,VECTOR,EIGVAL,NITER, HM141789
     1                NITRSP, NITROP, EPSP, EPOP, IR, US, H, MAXP, NC,  HM141790
     2                AITKEN, NAKSR, NAKDR, NTAPE3 )                    HM141791
      IF  ( NAERO )        3584,3582,3584                               HM141792
 3582 WRITE TAPE NTAPE7,  ((VECTOR(I,J),I=1,NFLEX),J=1,MODES)           HM141793
      REWIND NTAPE7                                                     HM141794
                                                                        HM141795
 3584 CONTINUE                                                          HM141796
      REWIND NTAPE8                                                     HM141797
      WRITE OUTPUT TAPE NTAPE3, 200                                     HM141798
      MODES2=NC*MODES                                                   HM141799
      DO  364  I=1,MODES2,NC                                            HM141800
        J=I/NC+NC-1                                                     HM141801
      IF  ( EIGVAL(I) )      359,360,360                                HM141802
  359 WRITE OUTPUT TAPE NTAPE3, 208, J                                  HM141803
      OMEGA(J)=0.                                                       HM141804
      GOTO  361                                                         HM141805
  360   CMEGA(J)=1. / SQRTF( EIGVAL(I) )                                HM141806
        DAMP=0.                                                         HM141807
        VELC=0.                                                         HM141808
  361   GOTO  (364,362),NC                                              HM141809
  362   CAMP= EIGVAL(I+1) / EIGVAL(I)                                   HM141810
        VELC = PICON*OMEGA(J)*BREF*VELCTY(NVEL)                         HM141811
  364   WRITE OUTPUT TAPE NTAPE3, 201, J, OMEGA(J), OAMP, VELC          HM141812
      IF  ( NPUNCH )         366,368,366                                HM141813
  366 WRITE OUTPUT TAPE NTAPEO, 205, NSIZES(1), MODES,BCDZ,NCAROS       HM141814
      NCRDS=NCARDS+1                                                    HM141815
```

```
      CALL MPUNCH (OMEGA,J,1,0,1,1,BCDZ,MAXR,NTAPEO,NCRDS )             HM141816
      WRITE OUTPUT TAPE NTAPE3, 206, 8CDZ, NCARDS, BCDZ, NCRDS          HM141817
      NCARDS=NCRDS+1                                                    HM141818
  368 L=1                                                               HM141819
      K1=0                                                              HM141820
C***********************************************************************HM141821
      WRITE OUTPUT TAPE NTAPE3, 203                                     HM141822
      DO  378  ISUR=1,NSUR                                              HM141823
      IF  ( NRIGID )          371,369,371                               HM141824
  369 READ TAPE NTAPE5, NSIZE, NMODES,((HF1(I,J),I=1,NSIZE),J=1,NMODES) HM141825
      CALL DPMLTX (HF1,1,VECTOR,NC,HF,NSIZE,NMODES,MODES,MAXR,MAXP,     HM141826
     1             MAXR,1)                                              HM141827
      NCMODE=NC*NMODES                                                  HM141828
      GOTO  375                                                         HM141829
  371 READ TAPE NTAPE9, NSIZE, I,((H1(I,J),I=1,NSIZE),J=1,NR2)          HM141830
      READ TAPE NTAPE5, I,NMODES, ((HF(I,J),I=1,NSIZE), J=1,NMODES)     HM141831
      NCMODE=NC*NMODES                                                  HM141832
      CALL DPMLTX (H1,NC,HRF,NC,HF1,NSIZE,NRIGID,NFLEX,MAXR,MAXQ,MAXR,1)HM141833
        DO  374  I=1,NSIZE                                              HM141834
        CO  372  J=1,NFLEX2                                             HM141835
  372     HF1(I,J)=-HF1(I,J)                                            HM141836
        DO  374  J=1,NCMODE,NC                                          HM141837
          K= J/NC +NC -1                                                HM141838
          L=K1+J                                                        HM141839
  374     HF1(I,L)=HF1(I,L)+HF(I,K)                                     HM141840
       IF  ( NFREE )          3742,3741,3742                            HM141841
 3741 K1=K1+NCMODE                                                      HM141842
 3742 CALL DPMLTX (HF1,NC,VECTOR,NC,HF,NSIZE,NFLEX,MODES,MAXR,          HM141843
     1             MAXP,MAXR,1)                                         HM141844
  375 DO  376  I=1,MODES2,NC                                            HM141845
        INDEX=0                                                         HM141846
        K=ISUR*NC-NC+1                                                  HM141847
        L=I/NC+NC-1                                                     HM141848
  376   CALL NPNRMX (HF(1,I),HF(1,I),NSIZE,US(K,L),INDEX,MAXR,NC,1)     HM141849
       IF  ( NSUR-1 )         378,377,378                               HM141850
  377 DO  3775   L=1,MODES2,NC                                          HM141851
        I=L/NC +NC -1                                                   HM141852
        H(1,I)=US(1,I)                                                  HM141853
```

```
3775   H(2,I)=US(2,I)                                                   HM141854
     GOTO  401                                                          HM141855
 378 WRITE TAPE NTAPE8, NSIZE, MODES2, ((HF(I,J),I=1,NSIZE),J=1,MODES2)HM141856
     REWIND NTAPE8                                                      HM141857
     DO  400  L=1,MODES2,NC                                             HM141858
       I=L/NC + NC - 1                                                  HM141859
     NAKOR(I)=1                                                         HM141860
       GOTO  (384,392),NC                                               HM141861
 384   H(1,I)=US(1,I)                                                   HM141862
       DO  390  ISUR=2,NSUR                                             HM141863
         IF  ( H(1,I)-US(ISUR,I))  388,390,390                          HM141864
 388     H(1,I)=US(ISUR,I)                                              HM141865
     NAKOR(I)=ISUR                                                      HM141866
 390     CONTINUE                                                       HM141867
       GOTO  400                                                        HM141868
 392 INDEX=1                                                            HM141869
     H(1,I)=US(1,I)**2+US(2,I)**2                                       HM141870
     J=2*NSUR                                                           HM141871
     DO  398  ISUR=3,J,2                                                HM141872
       IF  ( H(1,I)-(US(ISUR,I)**2+US(ISUR+1,I)**2) )     396,398,398  HM141873
 396   INDEX=ISUR                                                       HM141874
       H(1,I)=(US(ISUR,I)**2+US(ISUR+1,I)**2)                           HM141875
     NAKOR(I)=ISUR/2+1                                                  HM141876
 398   CONTINUE                                                         HM141877
 399   H(1,I)=US(INDEX,I)                                               HM141878
       H(2,I)=US(INDEX+1,I)                                             HM141879
 400 CONTINUE                                                           HM141880
                                                                        HM141881
 401 DO 414  ISUR=1,NSUR                                                HM141882
     IF  ( NSUR-1 )         4011,4012,4011                              HM141883
4011   READ TAPE NTAPE8, NSIZE, MODES2, ((HF(I,J),I=1,NSIZE),J=1,       HM141884
    1                    MODES2 )                                       HM141885
4012 DO  403   L=1,MODES2,NC                                            HM141886
       I=L/NC+NC-1                                                      HM141887
       IF  ( NSUR-1 )       4018,4016,4018                              HM141888
4018   IF  ( NAKOR(I)-ISUR )   402,403,402                              HM141889
 402 GOTO  (4021,4022),NC                                               HM141890
4021 GUESS(L,1)=H(1,I) / US(ISUR,I)                                     HM141891
```

```
      GOTO  4023                                                        HM141892
 4022 GUESS(L,1)=( H(1,I)*US(2*ISUR-1,I)+H(2,I)*US(2*ISUR,I) ) /        HM141893
     1             ( US(2*ISUR-1,I)**2 + US(2*ISUR,I)**2 )               HM141894
      GUESS(L+1,1)=( H(2,I)*US(2*ISUR-1,I) - H(1,I)*US(2*ISUR,I) ) /    HM141895
     1             ( US(2*ISUR-1,I)**2 + US(2*ISUR,I)**2 )               HM141896
 4023 CALL NPNRMX (HF(1,L),HF(1,L),NSIZE,GUESS(L,1),-1,MAXR,NC,1 )      HM141897
 4016   GUESS(I,1)=H(1,I)                                               HM141898
        GUESS(I,2)= H(2,I)                                              HM141899
  403 CONTINUE                                                          HM141900
  405 WRITE OUTPUT TAPE NTAPE3, 204, ISUR, NSIZE                        HM141901
        CALL MPRINT  (HF,NSIZE,MODES2,MAXR,NTAPE3)                      HM141902
                                                                        HM141903
      IF ( NPUNCH )           406,414,406                               HM141904
  406 IF  ( ISUR-1 )          408,410,408                               HM141905
  408 WRITE OUTPUT TAPE NTAPE0, 205, NSIZE, MODES2, BCDZ, NCARDS        HM141906
      NCRDS=NCARDS+1                                                    HM141907
      GOTO  412                                                         HM141908
  410 NCRDS=NCARDS                                                      HM141909
  412 CALL MPUNCH  (HF,NSIZE,MODES2,0,1,1,BCDZ,MAXR,NTAPE0,NCRDS)       HM141910
      WRITE OUTPUT TAPE NTAPE3, 206, BCDZ, NCARDS, BCDZ, NCRDS          HM141911
      NCARDS=NCRDS+1                                                    HM141912
  414 CONTINUE                                                          HM141913
                                                                        HM141914
  450 CONTINUE                                                          HM141915
      IF  ( NAERO )           480,460,480                               HM141916
  460 IF  ( NRIGID)           461,462,461                               HM141917
  461 READ TAPE NTAPE7,     ((HF(I,J),I=1,NRIGID),J=1,NFLEX)            HM141918
  462 READ TAPE NTAPE7,   ((HFF(I,J),I=1,NFLEX),J=1,NFLEX)              HM141919
      READ TAPE NTAPE7,   ((US(I,J),I=1,NFLEX),J=1,MODES)               HM141920
      IF  ( NRIGID )          463,467,463                               HM141921
  463 DO  464  I=1,NRIGID                                               HM141922
        DO  464  J=1,NFLEX                                              HM141923
  464     HF(I,J)=-HF(I,J)                                              HM141924
                                                                        HM141925
      CALL DPMLTX   (HF,1,US,1,HRRR ,NRIGID,NFLEX,MODES,MAXR,MAXP,MAXQ, HM141926
     1               1)                                                 HM141927
      CALL DPMLTX   (HRF,1,US,1,HF,NRIGID,NFLEX,MODES,MAXQ,MAXP,MAXR,1) HM141928
                                                                        HM141929
```

```
    DO  465  I=1,NRIGID
      DO  465  J=1,MODES
465     B(J,I) = HRRR (I,J)

    CALL DPMLTX  (B,1,HF,1,H,MODES,NRIGID,MODES,MAXP,MAXR,MAXP,1)


    DO  466  I=1,MODES
      DO  466  J=1,MODES
466     HF(I,J) = H(J,I)

467 CALL DPMLTX  (HFF,1,US,1,H,NFLEX,NFLEX,MODES,MAXP,MAXP,MAXP,1)

    DO  468   I=1,NFLEX
      DO  468   J=1,MODES
468     HFF(J,I)=US(I,J)

    CALL DPMLTX  (HFF,1,H,1,US,MODES,NFLEX,MODES,MAXP,MAXP,MAXP,1)
    IF  ( NRIGID )          469,472,469
469 DO  470  I=1,MODES
      DO  470  J=1,MODES
470     US(I,J) = US(I,J) +HF(I,J)

472 DO  474  I=1,MODES
474   US(I,I) = US(I,I)  /(GUESS(I,1)**2)

    WRITE OUTPUT TAPE NTAPE3, 209, (I,US(I,I),I=1,MODES)

480 CONTINUE

    MAXR = 50
    NGC= 98765

    CALL CHAIN (1,NTAPE1)

    END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
2598	05046	16936	41050

STORAGE LOCATIONS FOR VARIABLES APPEARING IN COMMON STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
AITKEN	32044	76454	ARF	27141	65005	A	32041	76451	BARMRR	32187	76673
BOOLT	32473	77331	BREF	32411	77233	BR	32409	77231	8	17596	42274
OMBAR	32259	77003	DM	32367	77157	DQ	32331	77113	EIGVAL	17036	41214
EPOP	32047	76457	EPSP	32048	76460	FREQ	32451	77303	GUESS	17116	41334
H1	21941	52665	HF1	32041	76451	HFF	32041	76451	HFR	23941	56605
HF	27141	65005	HRF	17596	42274	HRRR	19621	46245	HRR	18661	44345
H	21941	52665	ISW	32541	77435	ISXST	32561	77461	IT	32561	77461
MAXP	32480	77340	MAXQ	32482	77342	MAXR	32483	77343	MAXS	32481	77341
MODE	32486	77346	NAERO	32115	76563	NAKDR	16961	41101	NAKSR	17621	42325
NCANT	32489	77351	NCARDS	32049	76461	NC	32476	77334	NDENS	32487	77347
NFLEX2	32477	77335	NFLEX	32484	77344	NFREE	32490	77352	NFUS	32488	77350
NGO	32475	77333	NITER	16986	41132	NITRDP	32045	76455	NITRSP	32046	76456
NPOINT	32485	77345	NPUNCH	32474	77332	NR2	32478	77336	NRHO	32114	76562
NRIGIO	32491	77353	NSIZES	32521	77411	NSRFLX	32410	77232	NSURFS	32479	77337
NSUR	32492	77354	NTAPEO	32050	76462	NTAPE1	32501	77365	NTAPE2	32500	77364
NTAPE3	32499	77363	NTAPE4	32498	77362	NTAPE5	32497	77361	NTAPE6	32496	77360
NTAPE7	32495	77357	NTAPE8	32494	77356	NTAPE9	32493	77355	NVEL	32051	76463
OMEGA	17621	42325	PICON	32113	76561	RHO	32471	77327	SPRING	32112	76560
S	32388	77204	US	23941	56605	VECTOR	19621	46245	VELCTY	32072	76510

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
8COZ	2597	05045	DAMP	2596	05044	FFM	2595	05043	INDEX	2594	05042
IR	2593	05041	I	2592	05040	ISUR	2591	05037	J	2590	05036
K1	2589	05035	K	2588	05034	L	2587	05033	MODES2	2586	05032
MODES	2585	05031	NAERO	2584	05030	NCMODE	2583	05027	NCRDS	2582	05026
NMODES	2581	05025	N	2580	05024	NSIZE	2579	05023	VELC	2578	05022

SYM8DLS AND LDCATIDNS FOR SOURCE PROGRAM FDRMAT STATEMENTS

	EFN	LOC		EFN	LDC		EFN	LOC		EFN	LOC
8)68	200	04757	8)69	201	04736	8)6A	202	04731	8)6B	203	04635
8)6C	204	04625	8)6D	205	04613	8)6E	206	04607	8)6F	207	04571
8)6G	208	04563	8)6H	209	04551						

LDCATIONS FDR DTHER SYM8OLS NOT APPEARING IN SDURCE PRDGRAM

	DEC	DCT		DEC	OCT		DEC	OCT		DEC	DCT
1)	2544	04760	2)	2372	04504	3)	2381	04515	4)	32767	77777
6)	2384	04520	A)1C2	2270	04336	A)103	2283	04353	A)106	2296	04370
A)107	2309	04405	A)109	2322	04422	A)10A	2331	04433	A)108	2344	04450
A)10D	2357	04465	C)G0	2548	04764	C)G1	2549	04765	C)G2	2550	04766
C)G3	2551	04767	C)G4	2552	04770	C)101	2553	04771	C)102	2554	04772
C)103	2555	04773	C)105	2556	04774	C)106	2557	04775	C)107	2558	04776
C)109	2559	04777	C)10A	2560	05000	C)108	2561	05001	C)10C	2562	05002
C)10D	2563	05003	C)10E	2564	05004	C)10F	2565	05005	C)200	2566	05006
C)201	2567	050C7	C)202	2568	05010	C)203	2569	05011	C)204	2570	05012
C)205	2571	05013	C)206	2572	05014	C)207	2573	05015	C)208	2574	05016
C)209	2575	05017	C)20A	2576	05020	C)208	2577	05021	D)111	253	00375
D)11F	415	00637	D)14J	1420	02614	D)14V	1547	03013	D)15H	1746	03322
D)15I	1750	03326	D)22Q	725	01325	D)23K	973	01715	D)24T	1536	03000
D)252	1589	03065	D)254	1623	03127	D)25J	1753	03331	D)25L	1794	03402
D)25M	1822	03436	D)34T	1535	02777	D)352	1588	03064	D)35H	1745	03321
D)35L	1793	03401	D)35M	1821	03435	D)401	32	00040	D)447	1261	02355
D)44A	1276	02374	D)44F	1341	02475	D)44G	1355	02513	D)458	1638	03146
D)45N	1830	03446	D)47B	2251	04313	D)547	1260	02354	D)65N	1829	03445
D)75H	1744	03320	E)2P	705	01301	E)3G	930	01642	E)3J	960	01700
E)3N	1043	02023	E)4U	1541	03005	E)53	1600	03100			

LDCATIONS OF NAMES IN TRANSFER VECTDR

	CEC	DCT		DEC	DCT		DEC	DCT		DEC	DCT
CHAIN	14	00016	DPMLTX	4	00004	MITERS	10	00012	MPRINT	9	00011
MPUNCH	12	00014	NPNRMX	13	00015	SQRT	11	00013	(FIL)	8	00010
(FPT)	0	00000	(RLR)	3	00003	(RWT)	1	00001	(STB)	5	00005
(STH)	7	00007	(TSH)	2	00002	(WLR)	6	00006			

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

CHAIN	OPMLTX	MITERS	MPRINT	MPUNCH	NPNRMX	SQRT	(FIL)
(FPT)	(RLR)	(RWT)	(STB)	(STH)	(TSB)	(WLR)	

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
301	27	00066	302	30	00121	304	32	00126	290	36	00155
294	46	00220	305	60	00315	306	61	00322	307	62	00333
297	64	00352	298	66	00365	296	68	00376	308	83	00523
309	84	00531	310	98	00650	311	100	00663	312	102	00700
314	104	00706	316	115	00763	318	127	01055	320	139	01145
322	140	01155	324	152	01242	326	160	01302	350	163	01326
351	164	01332	352	172	01370	354	175	01412	355	180	01453
356	182	01462	358	185	01476	3582	188	01535	3584	197	01577
359	204	01644	360	208	01661	361	211	01675	362	212	01702
364	214	01716	366	217	01743	368	225	02024	369	230	02050
371	243	02136	372	266	02356	374	270	02425	3741	272	02452
3742	273	02455	375	275	02476	376	279	02545	377	282	02604
3775	285	02642	378	287	02650	384	301	02755	388	304	03006
390	306	03014	392	308	03021	396	313	03101	398	316	03130
399	317	03135	400	319	03141	401	320	03147	4011	322	03206
4012	331	03250	4018	334	03323	402	335	03327	4021	336	03332
4022	338	03340	4023	340	03403	4016	342	03437	403	344	03447
405	345	03456	406	350	03502	408	351	03505	410	355	03531
412	356	03533	414	361	03573	450	362	03606	460	364	03616
461	365	03620	462	373	03656	463	390	03754	464	392	03767
465	399	04050	466	404	04117	467	405	04131	468	409	04166
469	413	04221	470	415	04241	472	416	04254	474	417	04260
480	423	04314									

```
ENTRY POINTS TO SUBROUTINES REQUESTED FROM LIBRARY,   CHAIN     1 ,  3
     (TSHM)      (RTN)      (STHM)      (FIL)      EXIT      (IOS)      (RDS)      (RCH)      (TCO)
     (TRC)       (TEF)      (BSR)       (WRS)      (FPT)     SETH       (RWT)      (STB)      (WLR)
     (TSB)       (RLR)      CHAIN

ENTRY POINTS TO SUBROUTINES REQUESTED FROM LIBRARY,   CHAIN     2 ,  3
     (IOS)       (WRS)      (RCH)       (WTC)      (WER)     (TES)      (STHM)     (FIL)      (DFMP)
     (DFAD)      (DFDP)     DEXP(2      (DFSB)     (RWT)     (TSB)      (RLR)      (STB)      (WLR)
     SQRT        DSQRT      (FPT)       CHAIN

        EXECUTICN 16.546
```

```
                                                               3/11/63      PAGE 1

      SUBROUTINE  RDLN  (NTAPE2, NTAPE3, I )                  HM050001
    1 FORMAT(80H                                              HM050002
     1                          )                             HM050003
    2 FORMAT(1H1)                                             HM050004
    3 FORMAT  ( 1H0 )                                         HM050005
                                                              HM050006
      READ INPUT TAPE NTAPE2, 1                               HM050007
      GOTU   (4,5),I                                          HM050008
                                                              HM050009
    4 WRITE OUTPUT TAPE NTAPE3, 2                             HM050010
      GOTO  6                                                 HM050011
                                                              HM050012
    5 WRITE OUTPUT TAPE NTAPE3, 3                             HM050013
                                                              HM050014
    6 WRITE OUTPUT TAPE NTAPE3, 1                             HM050015
                                                              HM050016
      RETURN                                                  HM050017
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

```
                                       STORAGE NOT USED BY PROGRAM

        DEC   OCT              DEC   OCT
         79 00117            32561 77461

                          SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

        EFN   LOC              EFN   LOC              EFN   LOC              EFN   LOC
  8)1     1 00115      8)2       2 00076      8)3       3 00075

                         LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

        DEC   OCT              DEC   OCT              DEC   OCT              DEC   OCT
   6)    54 00066      C)GO     78 00116      E)1      30 00036

                                     LOCATIONS OF NAMES IN TRANSFER VECTOR

        DEC   OCT              DEC   OCT              DEC   OCT              DEC   OCT
 (FIL)    3 00003     (RTN)      1 00001     (STH)      2 00002     (TSH)      0 00000

                              ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

(FIL)       (RTN)       (STH)       (TSH)

                EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

 EFN   IFN   LOC         EFN   IFN   LOC         EFN   IFN   LOC         EFN   IFN   LOC
   4     8 00037           5    10 00046           6    11 00054
```

```
      SUBROUTINE MMULTO  (A,N1,B,N2,C,M,N,K,MA,MB,MC)                    HM050561
                                                                         HM050562
      OIMENSION  A(1), 8(1), C(1)                                        HM050563
                                                                         HM050564
      IC=1                                                               HM050565
      IA=MC*K                                                            HM050566
      IB=MA*N                                                            HM050567
      ID=MA                                                              HM050568
      IH=MC                                                              HM050569
      IJ=MC                                                              HM050570
                                                                         HM050571
      IF ( N1 )  4,3,4                                                   HM050572
    3 IF ( N2 )  7,8,7                                                   HM050573
    4 IB=2*IB                                                            HM050574
      ID=2*IO                                                            HM050575
      IF ( N2 )  5,6,5                                                   HM050576
    5 IC=2                                                               HM050577
      GOTO  7                                                            HM050578
    6 IH=2*IH                                                            HM050579
      IC=3                                                               HM050580
    7 IA=2*IA                                                            HM050581
      IJ=2*IJ                                                            HM050582
    8 00  18  I=1,M                                                      HM050583
      INC=0                                                              HM050584
      00  11  J=I,IA,IH                                                  HM050585
        C(J)=0.                                                          HM050586
        IN=INC                                                           HM050587
        00  10  L=I,IB,ID                                                HM050588
          IN=IN+1                                                        HM050589
   10     C(J)=C(J)+A(L)*B(IN)                                           HM050590
   11   INC=INC+M8                                                       HM050591
      INC=0                                                              HM050592
      GOTO  (18,12,15),IC                                                HM050593
   12 00  14  J=I,IA,IJ                                                  HM050594
        IE=I+MA                                                          HM050595
        IF=J+MC                                                          HM050596
        IN=INC                                                           HM050597
        DO  13  L=IE,IB,ID                                               HM050598
```

```
      IN=IN+1                                   HM050599
      IG=IN+MB                                  HM050600
      C(IF)=C(IF)+A(L)*B(IN)                    HM050601
13    C(J)=C(J)-A(L)*B(IG)                      HM050602
14   INC=INC+MB *2                              HM050603
     GOTO  18                                   HM050604
15 IE=I+MC                                      HM050605
   IF=I+MA                                      HM050606
   DO  17  J=IE,IA,IJ                           HM050607
     IN=INC                                     HM050608
     C(J)=0.                                    HM050609
     DO  16  L=IF,IB,ID                         HM050610
       IN=IN+1                                  HM050611
16     C(J)=C(J)+A(L)*B(IN)                     HM050612
17   INC=INC+MB                                 HM050613
18 CONTINUE                                     HM050614
   RETURN                                       HM050615
   END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)           HM050616
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
309	00465	32561	77461

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSICN, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
IA	308	00464	IB	307	00463	IC	306	00462	ID	305	00461
IE	304	00460	IF	303	00457	IG	302	00456	IH	301	00455
IJ	300	00454	INC	299	00453	IN	298	00452	I	297	00451
J	296	00450									

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
2)	278	00426	3)	282	00432	6)	283	00433	9)	289	00441
C)G0	290	00442	C)G2	291	00443	C)G3	292	00444	C)G4	293	00445
C)G5	294	00446	C)200	295	00447	D)20D	174	00256	D)407	123	00173
D)40M	268	00414	E)1	81	00121						

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
3	11	00122	4	12	00126	5	15	00140	6	17	00145
7	19	00155	8	21	00165	10	28	00223	11	29	00231
12	32	00246	13	40	00317	14	41	00327	15	43	00341
16	50	00377	17	51	00405	18	52	00415			

```
      SUBROUTINE MPRINT (A,M,N,MD,NTAPE)

      DIMENSION   A(1), IT(6), C(6)
      EQUIVALENCE  (IT,C)

    2 FORMAT   (1H , 4X, 6( 6X, 7HCOLUMN   1I4  )   ///    )
    3 FORMAT   (1H  1I4, X,   (6E 17.8)  )

      N1=N
      N2=6
      N3=6
      N4=1
    4 IF (N3-N1)  6,6,5
    5 N2=N1-N3+6
      N3=N1
    6 K=0
      DO  7  I= N4,N3
        K=K+1
    7   IT(K)=I
      WRITEOUTPUTTAPE NTAPE, 2,  (IT(I),I=1,N2)
      DO  9  I=1,M
        K=0
        L=MD*(N4-1)+I
        DO  8  J=N4,N3
          K=K+1
          C(K)=A(L)
    8     L=L+MD
    9   WRITEOUTPUTTAPE NTAPE, 3, I,  (C(K),K=1,N2)
      IF (N3-N1) 10,11,11
   10 N3=N3+6
      N4=N4+6
      GOTO  4
   11 RETURN
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

```
                                  STORAGE NOT USED BY PROGRAM

       DEC   OCT               DEC   OCT
       192 00300             32561 77461

                      STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

       DEC   OCT               DEC   OCT               DEC   OCT               DEC   OCT
  C    191 00277       IT      191 00277

 STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

       DEC   OCT               DEC   OCT               DEC   OCT               DEC   OCT
  I    185 00271       K       184 00270       L       183 00267       N1      182 00266
 N2    181 00265       N3      180 00264       N4      179 00263

                           SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

       EFN   LOC               EFN   LOC               EFN   LOC               EFN   LOC
 8)2     2 00254       8)3       3 00244

                           LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

       DEC   OCT               DEC   OCT               DEC   OCT               DEC   OCT
  1)   174 00256        2)     151 00227        6)     154 00232        9)     173 00255
C)G0   176 00260      C)G2     177 00261      C)202    178 00262       E)E     139 00213

                                  LOCATIONS OF NAMES IN TRANSFER VECTOR

       DEC   OCT               DEC   OCT               DEC   OCT               DEC   OCT
(FIL)    1 00001      (STH)      0 00000

                             ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

(FIL)      (STH)
```

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
4	11	00040	5	12	00045	6	14	00053	7	17	00071
8	29	00153	9	30	00161	10	37	00214	11	40	00223

```
C  MATRIX READ SUBROUTINE                                               HM050054
C     CALL MREAD    (A,M,N,IFDRM,IRDW,ITRA,IORG,T,MD,NTAPE2,NTAPE3     ) HM050055
C                                                                        HM050056
C          A = MATRIX TO READ IN       ITRA = 0, TRA CARD AFTER MATRIX  HM050057
C          M = NUMBER OF RDWS                 =+1, TRA CARD AFTER EACH RDWHM050058
C          N = NUMBER OF CDLUMNS                         (OR COLUMN )   HM050059
C          IFDRM = -1, FORMAT(12A6)    IORG = DRIGIN OF FIRST C.B. CARD HM050060
C                =  0, COLUMN 8INARY   T =MDXN TEMPORARY CELLS          HM050061
C                = +1, FDRMAT(6E12.8)  MD= DIMENSIONED NUMBER OF ROWS   HM050062
C          IRDW = .0, MATRIX BY CDLUMNS                          IN A   HM050063
C               = +1, MATRIX BY ROWS   NTAPE2 = INPUT TAPE              HM050064
C                                      NTAPE3 = OUTPUT TAPE             HM050065
                                                                         HM050066
      SUBROUTINE MREAD  (A,M,N,IFORM,IROW,ITRA,IORG,T,MD,NTAPE2,NTAPE3 )HM050067
      DIMENSION    A(1), T(1)                                           HM050068
                                                                         HM050069
    1 FORMAT   (6E12.8)                                                 HM050070
    2 FORMAT   (12A6)                                                   HM050071
    3 FORMAT   ( // 26H THATS ALL YOUR DATA.               )            HM050072
      MN=MD*N                                                           HM050073
      DO  5  I=1,MN                                                     HM050074
        T(I)=0.                                                         HM050075
    5   A(I)=0.                                                         HM050076
                                                                         HM050077
      IF ( IFORM )  39,15,6                                             HM050078
    6 IF  ( IROW )   8,7,8                                              HM050079
    7 K3=1                                                              HM050080
      K4=N                                                              HM050081
      K5=MD                                                             HM050082
      K6=M-1                                                            HM050083
       K2=1                                                             HM050084
      GOTO 9                                                            HM050085
    8 K2=MN                                                             HM050086
      K3=MD                                                             HM050087
      K4=M                                                              HM050088
      K5=1                                                              HM050089
      K6=0                                                              HM050090
```

```
 9 DO  11  I=1,K4                                          HM050091
     K1=I*K5-K5+1                                          HM050092
     IF (K6)   10,11,10                                    HM050093
10   K2=K1+K6                                              HM050094
11   READ INPUT TAPE  NTAPE2, 1,  (A(L),L=K1,K2,K3)        HM050095
     GOTO 36                                               HM050096
15 K1=N                                                    HM050097
   K2=M                                                    HM050098
   K3=1                                                    HM050099
   IF ( IORG-1 )  16,17,17                                 HM050100
16 K3=2                                                    HM050101
17 IF  ( IROW )  18,19,18                                  HM050102
18 K2=N                                                    HM050103
   K1=M                                                    HM050104
19 IF ( ITRA )    22,21,22                                 HM050105
21 K1=1                                                    HM050106
22 K=0                                                     HM050107
   DO  23  I=1,K1                                          HM050108
     K4=K+K3                                               HM050109
     K5=1                                                  HM050110
   CALL  BINRD  (T(K4), K5, L, NTAPE2, NTAPE3 )            HM050111
     GOTO  (23,38,23,23),L                                 HM050112
23 K=K+K2                                                  HM050113
                                                           HM050114
   IF  ( IROW )  28,24,28                                  HM050115
24 L=0                                                     HM050116
   IF  ( IORG-1)  26,26,25                                 HM050117
25 L=IORG-1                                                HM050118
26 DO  27  I=1,N                                           HM050119
     J=I*MD-MD                                             HM050120
     DO  27  K=1,M                                         HM050121
       J=J+1                                               HM050122
   L=L+1                                                   HM050123
27 A(J)=T(L)                                               HM050124
   GOTO  36                                                HM050125
                                                           HM050126
28 L=0                                                     HM050127
   IF  (IORG-1)  30,30,29                                  HM050128
```

```
29 L=IORG-1                                           HM050129
30 DO  31  K=1,N                                      HM050130
     J= K*MD-MD                                       HM050131
     DO  31  I=K,MN,N                                 HM050132
       J=J+1                                          HM050133
       K1=L+I                                         HM050134
31     A(J)=T(K1)                                     HM050135
                                                      HM050136
36 RETURN                                             HM050137
                                                      HM050138
38 WRITE OUTPUT TAPE  NTAPE3, 3                       HM050139
   CALL EXIT                                          HM050140
   STOP                                               HM050141
39 READ INPUT TAPE NTAPE2, 2,  (A(I),I=1,M)           HM050142
   GOTO  36                                           HM050143
   END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NOT USED BY PRCGRAM

DEC	OCT	DEC	OCT
406	00626	32561	77461

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
I	405	00625	J	404	00624	K1	403	00623	K2	402	00622
K3	401	00621	K4	400	00620	K5	399	00617	K6	398	00616
K	397	00615	L	396	00614	MN	395	00613			

SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

	EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC
8)1	1	00601	8)2	2	00577	8)3	3	00575			

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
2)	363	00553	3)	366	00556	6)	367	00557	9)	386	00602
C)G1	388	00604	C)G2	389	00605	C)G3	390	00606	C)G4	391	00607
C)200	392	00610	C)203	393	00611	C)205	394	00612	D)107	128	00200
D)212	301	00455	D)216	337	00521	E)E	170	00252	E)H	181	00265
E)J	191	00277	E)N	226	00342						

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
BINRO	2	00002	EXIT	5	00005	(FIL)	4	00004	(RTN)	1	00001
(STH)	3	00003	(TSH)	0	00000						

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

BINRD EXIT (FIL) (RTN) (STH) (TSH)

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
5	10	00124	6	12	00134	7	13	00140	8	19	00156
9	24	00172	10	27	00213	11	28	00216	15	34	00237
16	38	00253	17	39	00255	18	40	002 7	19	42	00266
21	43	00272	22	44	00300	23	51	00335	24	53	00347
25	55	00360	26	56	00365	27	61	00417	28	63	00431
29	65	00442	30	66	00447	31	71	00506	36	72	00522
38	74	00526	39	77	00537						

```
                          *     FIB II   FEBUARY 10,1961  ROGER ANDERSON                          HM050148
                          *     FIB II   APRIL 20,1961 REVISED ANDCORRECTED BY R. L. GAUTHIER     HM050149
                          **-*******************************************************************HM050150
                          * CALLING SEQUENCE                                                     HM050151
                          *       TSX      BINRD,4                                               HM050152
                          *       TSX      L(ARRAY)                                              HM050153
                          *       TSX      L(K1), WHERE K1 IS READ CONTROL.                      HM050154
                          *       TSX      L(K2), WHERE K2 IS ERROR CONTROL.                     HM050155
                          *       TSX      L(INPUT TAPE NUMBER)                                  HM050156
                          *       TSX      L(OUTPUT TAPE NUMBER)                                 HM050157
                          *                                                                      HM050158
                          *                K1 = ZERO, READ ONE CARD.                             HM050159
                          *                K1 = NONZERO, READ TO TRANSFER CARD.                  HM050160
                          *                                                                      HM050161
                          *       IF       K2 = 1, RECORD(S) READ CORRECTLY.                     HM050162
                          *                K2 = 2, END OF FILE ENCOUNTERED. READ END.            HM050163
                          *                K2 = 3, CHECKSUM ERROR OR TAPE CHECK.                 HM050164
                          *                K2 = 4, (WHEN K1=0) NEXT RECORD IS BCD.               HM050165
                          *********************************************************************HM050166
                          *                                                                      HM050167
                00011             ENTRY    BINRD                                                 HM050168
                          *                                                                      HM050169

 TRANSFER VECTOR
00000  743146623460      (IOS)
00001  745124623460      (RDS)
00002  745123303460      (RCH)
00003  746323463460      (TCO)
00004  746351233460      (TRC)
00005  746325263460      (TEF)
00006  742262513460      (BSR)
00007  256731636060      EXIT
00010  746651623460      (WRS)

00011  0634 00 4 00156   BINRD   SXA      BRDXR4,4             SAVE INDEX REGISTERS.           HM050170
00012  0634 00 2 00157           SXA      BRDXR2,2                                             HM050171
00013  0634 00 1 00160           SXA      BRDXR1,1                                             HM050172
00014  0500 60 4 00004           CLA*     4,4                                                  HM050173
```

```
00015  0622 00 0 00252        STD    BRDB5D                                     HM050174
00016  0500 60 4 00005        CLA*   5,4                                        HM050175
00017  0622 00 0 00251        STD    BRD6D                                      HM050176
00020 -0500 00 0 00245        CAL    BRD1D              SET K2=1. (AN ERROR     HM050177
00021  0602 60 4 00003        SLW*   3,4                WILL RESET K2.          HM050178
00022 -0500 00 0 00245        CAL    BRD1D                                      HM050179
00023  0602 00 0 00243        SLW    DNE                                        HM050180
00024 -0500 00 4 00001        CAL    1,4                INITIALIZE LOCATION     HM050181
00025  0400 00 0 00244        ADD    BRD1A                                      HM050182
00026  0621 00 0 00100        STA    BRDARY             DF ARRAY.               HM050183
                       ******************************************************************HM050184
                       *****  READ INPUT TAPE IN BINARY MDDE.                           HM050185
                       *      IF REDUNDANCY DCCURS, (1) THE RECDRD IS IN BCD MDDE AND ROUTINE  HM050186
                       *             WILL INDICATE K2 = 4, IF SINGLE READ DR K2 = 2 IF MULTI   HM050187
                       *             READ. (2) A LEGITIMATE REDUNDANCY HAS OCCURRED ON A       HM050188
                       *             BINARY RECDRD, A SECDND READ IS ATTEMPTED, IF REDUNDANCY  HM050189
                       *             PERSISTS A MESSAGE IS WRITTEN DN THE OUTPUT TAPE          HM050190
                       *             AND K2 IS SET TD 3.........                               HM050191
                       ******************************************************************HM050192
                       *                                                                HM050193
00027 -0500 00 0 00252 BRDRD  CAL    BRDB5D             ESTABLISH BINARY I-D    HM050194
00030  0074 00 4 00000        CALL   (IDS)              FDR INPUT TAPE.         HM050195
00031  0522 60 0 00001 BRDRD1 XEC*   $(RDS)             READ SELECT INPUT.      HM050196
00032 -0774 00 4 00257        AXC    BRDCMD,4                                   HM050197
00033  0522 60 0 00002        XEC*   $(RCH)             RESET AND LDAD CHANNEL  HM050198
00034 -0774 00 4 00035        AXC    *+1,4                                      HM050199
00035  0522 60 0 00003        XEC*   $(TCD)             DELAY                   HM050200
00036 -0774 00 4 00110        AXC    BRDTPC,4                                   HM050201
00037  0522 60 0 00004        XEC*   $(TRC)             TEST REDUNDANCY.        HM050202
00040 -0500 00 0 00245        CAL    BRD1D                                      HM050203
00041  0602 00 0 00243        SLW    DNE                                        HM050204
00042 -0774 00 4 00162 BRDTEF AXC    BRDEND,4                                   HM050205
00043  0522 60 0 00005        XEC*   $(TEF)             TEST END-OF-FILE.       HM050206
                       *                                                                HM050207
                       ******************************************************************HM050208
                       *****  TEST FDR TRANSFER CARD.                                   HM050209
                       *      IF TRANSFER CARD IS ENCDUNTERED, TERMINATE READ.          HM050210
                       ******************************************************************HM050211
```

```
00044 -0500 00 0 77740  BRDTCT CAL     BUFFER              IF DECREMENT (WORD COUNT)          HM050212
00045 -0320 00 0 00250         ANA     BROD37              IS ZERO, THIS IS                   HM050213
00046  0100 00 0 00152         TZE     BROTCO              A TRANSFER CARD........            HM050214
                        *****  TEST TO IGNORE CHECKSUM.                                       HM050215
00047 -0500 00 0 77740         CAL     BUFFER              IF COL 1, ROW 0                    HM050216
00050 -0320 00 0 00254         ANA     BROIGN              IS PUNCHED, IGNORE                 HM050217
00051 -0100 00 0 00071         TNZ     BRDMV               THE CHECKSUM.                      HM050218
                        *                                                                     HM050219
                        ************************************************************************HM050220
                        *****  ACCUMULATE AND CHECK THE CHECKSUM.  *****                      HM050221
                        *      IF CHECKSUM ERROR, WRITE MESSAGE, INDICATE                     HM050222
                        *              ERROR WITH 3 AT K2, AND CONTINUE.                      HM050223
                        *              (THE CARD WILL BE MOVED INTO THE ARRAY.)               HM050224
                        ************************************************************************HM050225
                        *                                                                     HM050226
00052 -0500 00 0 77740         CAL     BUFFER              SET UP NO. OF                      HM050227
00053 -0320 00 0 00250         ANA     BROD37                                                 HM050228
00054 -0734 00 4 00000         PDX     0,4                 WORDS FOR                          HM050229
00055  0634 00 4 00063         SXA     BINWDS,4            COMPUTING CHECKSUM                 HM050230
00056  1 00002 4 00057         TXI     *+1,4,2                                                HM050231
00057 -0634 00 4 00061         SXD     *+2,4                                                  HM050232
00060  0774 00 4 77740         AXT     BUFFER,4                                               HM050233
00061  1 00000 4 00062         TXI     *+1,4,**                                               HM050234
00062  0634 00 4 00065         SXA     BINCK,4                                                HM050235
00063  0774 00 4 00000  BINWDS AXT     **,4                ACCUMULATE ALL WORDS               HM050236
00064 -0500 00 0 77740         CAL     BUFFER              EXCEPT CHECKSUM WORD.              HM050237
00065  0361 00 4 00000   BINCK ACL     **,4                                                   HM050238
00066  2 00001 4 00065         TIX     *-1,4,1                                                HM050239
00067  0322 00 0 77741         ERA     BUFFER+1                                               HM050240
00070 -0100 00 0 00144         TNZ     BRDCSE              CHECKSUM ERROR                     HM050241
                        *                                                                     HM050242
                        ************************************************************************HM050243
                        *****  MOVE THE CARD IMAGE INTO USERS ARRAY.  *****                   HM050244
                        *      THE ARRAY IS FILLED IN FORTRAN FASHION. (BACKWARDS)            HM050245
                        *      THE INDEX NUMBER LOCATES THE FIRST WORD IN ARRAY.              HM050246
                        *      THE WORD COUNT IS USED TO TERMINATE THE MOVE.                  HM050247
                        *      BY DEFN...IF THE INDEX NUMBER = 1, PLACE FIRST WORD AT L(ARRAY)..HM050248
                        **-*********************************************************************HM050249
```

```
                                   *                                                       HM050250
00071 -0500 00 0 77740    BRDMV  CAL    BUFFER                                             HM050251
00072  0734 00 4 00000           PAX    0,4              INDEX ND. TD XR4.                 HM050252
00073 -0320 00 0 00250           ANA    BRDD37                                             HM050253
00074 -0737 00 2 00000           PDC    0,2              WDRD COUNT TO EXIT                HM050254
00075 -0634 00 2 00103           SXD    BRDMV1,2         TEST.                             HM050255
00076  0774 00 1 00000           AXT    0,1                                                HM050256
00077 -0500 00 1 77742           CAL    BUFFER+2,1                                         HM050257
00100  0602 00 4 00000    BRDARY SLW    **,4             **=(ARRAY+1)                      HM050258
00101  1 77777 1 00102           TXI    *+1,1,-1                                           HM050259
00102  1 00001 4 00103           TXI    *+1,4,1                                            HM050260
00103  3 00000 1 00077    BRDMV1 TXH    *-4,1,**         **=(WORD COUNT)                   HM050261
                          *                                                                HM050262
                          *********************************************************************HM050263
                          *****  TEST FOR MULTI OR SINGLE READ AND BRANCH ACCDRDINGLY.     HM050264
                          *********************************************************************HM050265
                          *                                                                HM050266
00104  0534 00 4 00156           LXA    BRDXR4,4                                           HM050267
00105  0520 60 4 00002           ZET*   2,4                                                HM050268
00106  0020 00 0 00031           TRA    BRDRD1           K1=NONZERO, READ                  HM050269
00107  0020 00 0 00156           TRA    BRDXR4           K1=ZERC, EXIT)))))                HM050270
                          *                                                                HM050271
                          ***********************************************Q*********************HM050272
                          *                                                                HM050273
                          *                                                                HM050274
                          *                                                                HM050275
00110  0522 60 0 00006    BRDTPC XEC*   $(BSR)           POSITICN FOR RETRY DR EXIT.       HM050276
00111 -0774 00 4 00256           AXC    BRDCSV,4         IF A BCD RECORD IS                HM050277
00112  0774 00 2 64000           AXT    26624,2          ENCDUNTERED, THE C(AR)            HM050278
00113 -0500 60 0 00002           CAL*   $(RCH)           DF CHANNEL WILL                   HM050279
00114  0602 00 0 00116           SLW    *+2              DENOTE ONLY 14 WDRCS              HM050280
00115 -0634 00 2 00116           SXD    *+1,2            TRANSMITTED. A BINARY             HM050281
00116  0 00000 0 00000           PZE                     RECDRD WILL BE DENCTED            HM050282
00117  0502 00 0 00256           CLS    BRDCSV           BY A GREATER NUMBER OF            HM050283
00120 -0320 00 0 00253           ANA    BRD5A7           WDRDS TRANSMITTED.                HM050284
00121  0402 00 0 00255           SUB    BUFORG            (THE SCHX INSTRUCTION IS ABSENT  HM050285
00122  0734 00 4 00000           PAX    0,4               FRDM IOS. MUST BUILD DNE.)       HM050286
00123  3 00021 4 00131           TXH    BRDBTC,4,17                                        HM050287
```

```
00124 -0500 00 0 00247        CAL    BRD3D          RECORD IS BCD MODE.        HM050288
00125  0534 00 4 00156        LXA    BRDXR4,4                                  HM050289
00126  0400 00 0 00245        ADD    BRD1D          SET K2=4 IF MULTI.         HM050290
00127  0602 60 4 00003        SLW*   3,4                                       HM050291
00130  0020 00 0 00157        TRA    BRDXR2         EXIT FROM BINRD.           HM050292
                       *                                                       HM050293
00131 -0500 00 0 00243 BRDBTC CAL    ONE            IF REDUNDANCY IS           HM050294
00132  0771 00 0 00001        ARS    1              CONSISTENT, REPORT         HM050295
00133  0602 00 0 00243        SLW    ONE            AND EXIT.                  HM050296
00134  0520 00 0 00243        ZET    UNE                                       HM050297
00135  0020 00 0 00031        TRA    BRDRD1                                    HM050298
00136  0534 00 4 00156        LXA    BRDXR4,4       SET K2=3.                  HM050299
00137 -0500 00 0 00247        CAL    BRD3D                                     HM050300
00140  0602 60 4 00003        SLW*   3,4                                       HM050301
00141  0774 00 1 00001        AXT    1,1            INDICATE BAD TAPE          HM050302
00142  0074 00 4 00173        TSX    BRDRPT,4       TO REPORTER.               HM050303
00143  0020 00 0 00042        TRA    BRDTEF         TRANSFER CARD TEST.        HM050304
                       *                                                       HM050305
00144  0774 00 1 00000 BRDCSE AXT    0,1            INDICATE CHECKSUM BUM      HM050306
00145  0534 00 4 00156        LXA    BRDXR4,4                                  HM050307
00146 -0500 00 0 00247        CAL    BRD3D                                     HM050308
00147  0602 60 4 00003        SLW*   3,4                                       HM050309
00150  0074 00 4 00173        TSX    BRDRPT,4       TO REPORTER.               HM050310
00151  0020 00 0 00071        TRA    BRDMV          MOVE.                      HM050311
                       *                                                       HM050312
00152  0534 00 4 00156 BRDTCD LXA    BRDXR4,4                                  HM050313
00153 -0500 00 0 77740        CAL    BUFFER         SET K1= TRANSFER ADDRESS   HM050314
00154  0767 00 0 00022        ALS    18             OF TRANSFER CARD......     HM050315
00155  0622 60 4 00002        STD*   2,4                                       HM050316
00156  0774 00 4 00000 BRDXR4 AXT    **,4                                      HM050317
00157  0774 00 2 00000 BROXR2 AXT    **,2                                      HM050318
00160  0774 00 1 00000 BROXR1 AXT    **,1                                      HM050319
00161  0020 00 4 00006        TRA    6,4                                       HM050320
                       *                                                       HM050321
00162  0502 00 0 00246 BRDEND CLS    BRD2D          THIS WILL RETURN           HM050322
00163  0522 60 0 00006        XEC*   $(BSR)         CONTROL TO USER            HM050323
00164  0020 00 0 00165        TRA    *+1            AT FIRST EOF, BUT          HM050324
00165  0760 00 0 00002        CHS                   WILL TAKE SYSTEM           HM050325
```

```
00166 -0625 00 0 00164          STL     *-2                 EXIT WHEN EOF IS                HM050326
00167 -0120 00 0 00007          TMI     $EXIT               ENCOUNTERED AGAIN....           HM050327
00170  0534 00 4 00156          LXA     BRDXR4,4            ....                            HM050328
00171  0602 60 4 00003          SLW*    3,4                 .....                           HM050329
00172  0020 00 0 00157          TRA     BRDXR2              ....                            HM050330
                         *                                                                  HM050331
                         ***********************************************************************HM050332
                         *****  INDEX REGISTER 1 DENOTES ERROR TYPE.                        HM050333
                         *              XR1=0, CHECKSUM ERROR                               HM050334
                         *              XR1=1, REDUNDANCY                                   HM050335
                         ***********************************************************************HM050336
                         *                                                                  HM050337
00173  0634 00 4 00203   BRDRPT SXA     RPTXR4,4                                            HM050338
00174 -0500 00 0 00251          CAL     BRD6D               ESTABLISH IO FOR                HM050339
00175  0074 00 4 00000          CALL    (IOS)               OUTPUT TAPE.                    HM050340
00176  0522 60 0 00010          XEC*    $(WRS)                                              HM050341
00177 -3 00000 1 00205          TXL     RPTCSE,1,0          TEST TYPE OF ERROR.             HM050342
00200 -0774 00 4 00260          AXC     REDCMD,4                                            HM050343
00201  0522 60 0 00002          XEC*    $(RCH)                                              HM050344
00202  0074 00 2 00221          TSX     RPTERR,2                                            HM050345
00203  0774 00 4 00000   RPTXR4 AXT     **,4                                                HM050346
00204  0020 00 4 00001          TRA     1,4                                                 HM050347
                         ***********************************************************************HM050348
                         *****  REPORT  THE CONTROL WORD OF CARD WITH BAD CHECKSUM.         HM050349
                         **-********************************************************************HM050350
                         *                                                                  HM050351
00205  0774 00 2 00002   RPTCSE AXT     2,2                 EDIT THE CONTROL                HM050352
00206  0560 00 0 77740          LDQ     BUFFER              WORD OF BAD CARD,               HM050353
00207  0774 00 1 00006   RPTECH AXT     6,1                 PLACE CONTROL WORD              HM050354
00210  0767 00 0 00003   RPTEDT ALS     3                   IN MESSAGE.                     HM050355
00211 -0763 00 0 00003          LGL     3                                                   HM050356
00212  2 00001 1 00210          TIX     RPTEDT,1,1                                          HM050357
00213  0602 00 2 00274          SLW     BRDCHK+2,2                                          HM050358
00214  2 00001 2 00207          TIX     RPTECH,2,1                                          HM050359
00215 -0774 00 4 00261          AXC     CSECMD,4                                            HM050360
00216  0522 60 0 00002          XEC*    $(RCH)                                              HM050361
00217  0074 00 2 00221          TSX     RPTERR,2            CHECK WRITE ERROR.              HM050362
00220  0020 00 0 00203          TRA     RPTXR4                                              HM050363
```

```
                                 *                                                            HM050364
00221  0634 00 4 00240  RPTERR SXA     BRDCOM,4          REMEMBER LAST COMMAND.              HM050365
00222 -0774 00 4 00223         AXC     *+1,4             DELAY AND TEST                      HM050366
00223  0522 60 0 00003         XEC*    $(TCO)            REDUNDANCY ON OUTPUT.               HM050367
00224 -0774 00 4 00231         AXC     IPTPCH,4                                              HM050368
00225  0522 60 0 00004         XEC*    $(TRC)                                                HM050369
00226 -0500 00 0 00252         CAL     BRDB5D            REESTABLISH IO FOR                  HM050370
00227  0074 00 4 00000         CALL    (IOS)             INPUT TAPE....                      HM050371
00230  0020 00 2 00001         TRA     1,2                                                   HM050372
00231  0522 60 0 00006  IPTPCH XEC*    $(BSR)            BACKSPACE OUTPUT                    HM050373
00232  0522 60 0 00010         XEC*    $(WRS)            TAPE AND ERASE.                     HM050374
00233 -0774 00 4 00234         AXC     *+1,4                                                 HM050375
00234  0522 60 0 00003         XEC*    $(TCO)                                                HM050376
00235 -0774 00 4 00237         AXC     *+2,4             TURN OFF POSSIBLE                   HM050377
00236  0522 60 0 00004         XEC*    $(TRC)            REDUNDANCY.                         HM050378
00237  0522 60 0 00010         XEC*    $(WRS)                                                HM050379
00240  0774 00 4 00000  BRDCOM AXT     **,4              RETRY WRITE.                        HM050380
00241  0522 60 0 00002         XEC*    $(RCH)                                                HM050381
00242  0020 00 0 00221         TRA     RPTERR                                                HM050382
                        *                                                                     HM050383
00243  0 00001 0 00000   ONE   PZE     0,0,1                                                 HM050384
00244  0 00000 0 00001  BRD1A  PZE     1                                                     HM050385
00245  0 00001 0 00000  BRD1D  PZE     0,0,1                                                 HM050386
00246  0 00002 0 00000  BRD2D  PZE     0,0,2                                                 HM050387
00247  0 00003 0 00000  BRD3D  PZE     0,0,3                                                 HM050388
00250  0 00037 0 00000  BRDD37 PZE     0,0,31                                                HM050389
00251  0 00003 0 00000  BRD6D  PZE     0,0,3                                                 HM050390
00252  0 00002 0 00020  BRDB5D PZE     16,0,2                                                HM050391
00253 +000000077777     BRD5A7 OCT     77777                                                 HM050392
00254  1 00000 0 00000  BRDIGN PON     0,0,0                                                 HM050393
00255  0 00000 0 77740  BUFORG PZE     BUFFER                                                HM050394
00256  0 00000 0 00000  BRDCSV PZE                                                           HM050395
                        *                                                                     HM050396
00257 -3 00033 0 77740  BRDCMD IOST    BUFFER,0,27                                           HM050397
00260  0 00005 0 00262  REDCMD IOCD    BRDMG1,0,5                                            HM050398
00261  0 00006 0 00267  CSECMD IOCD    BRDMG2,0,6                                            HM050399
                        *                                                                     HM050400
                        * MESSAGES                                                            HM050401
```

```
                                  *                                                      HM050402
00262  016321472560        BRDMG1 BCI     5,1TAPE CHECK ON INPUT IGNCRED.               HM050403
00263  233025234260
00264  464560314547
00265  646360312745
00266  465125243360
00267  012330252342        BRDMG2 BCI     3,1CHECKSUM ERRDR                             HM050404
00270  626444602551
00271  514651606060
00272  606060606060        BRDCHK BCI     3,                                            HM050405
00273  606060606060
00274  606060606060
                   77740   BUFFER EQU     -32                                           HM050406
                                  END                                                   HM050407
```

```
275 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
   243    ONE     23,    41,   131,   133,   134
     7   EXIT    167
    65  BINCK     62
    11  BINRD
   244  BRD1A     25
   245  BRD1D     20,    22,    40,   126
   246  BRD2D    162
   247  BRD3D    124,   137,   146
   251  BRO60     17,   174
    71  BROMV     51,   151
    27  BRDRD
     6  (BSR)    110,   163,   231
     0  (IOS)     30,   175,   227
     2  (RCH)     33,   113,   201,   216,   241
     1  (RDS)     31
     3  (TCO)     35,   223,   234
     5  (TEF)     43
     4  (TRC)     37,   225,   236
    10  (WRS)    176,   232,   237
    63 BINWDS     55
   253 BRO5A7    120
   100 BRDARY     26
   252 BRDB5D     15,    27,   226
   131 BRDBTC    123
   272 BRDCHK    213
   257 BRDCMO     32
   240 BRDCDM    221
   144 BRDCSE     70
   256 BRDCSV    111,   117
   250 BRDD37     45,    53,    73
   162 BRDEND     42
   254 BRDIGN     50
   262 BRDMG1    260
```

POST PROCESSOR ASSEMBLY DATA

```
  267 BRDMG2   261
  103 BRDMV1    75
   31 BRDRD1   106,  135
  173 BRDRPT   142,  150
  152 BRDTCD    46
   44 BRDTCT
   42 BRDTEF   143
  110 BRDTPC    36
  160 BRDXR1    13
  157 BRDXR2    12,  130,  172
  156 BRDXR4    11,  104,  107,  125,  136,  145,  152,  170
77740 BUFFER    44,   47,   52,   60,   64,   67,   71,   77,  153,  206,  255,  257,  275
  255 BUFORG   121
  261 CSECMD   215
  231 IPTPCH   224
  260 REDCMD   200
  205 RPTCSE   177
  207 RPTECH   214
  210 RPTEDT   212
  221 RPTERR   202,  217,  242
  203 RPTXR4   173,  220
```

NO ERROR IN ABOVE ASSEMBLY.

```
      SUBROUTINE  MNVRSX (AR,AI,B,C,KSZ,IGOOFD,NOP)                 HM050408
                                                                    HM050409
      DIMENSION AR(6,6), AI(6,6), B(6,6), C(6,6)                    HM050410
                                                                    HM050411
      IGOOFD=0                                                      HM050412
      IF ( NOP )      102,101,102                                   HM050413
  101 CALL INVERS  (AR,KSZ,IGOOFD)                                  HM050414
      GO TO 20                                                      HM050415
 102  CONTINUE                                                      HM050416
      DO 1 K=1,KSZ                                                  HM050417
      DO 1 L=1,KSZ                                                  HM050418
 1    B(K,L)=AR(K,L)                                                HM050419
      NO=0                                                          HM050420
      CALL INVERS(B,KSZ,NO)                                         HM050421
      IF (NO) 2,3,2                                                 HM050422
C  REAL MATRIX NOT SINGULAR                                         HM050423
C  MULT B*AI   STO. C                                               HM050424
 3    DO 4 K=1,KSZ                                                  HM050425
      DO 4 L=1,KSZ                                                  HM050426
      C(K,L)=0.0                                                    HM050427
      DO 4 I=1,KSZ                                                  HM050428
 4    C(K,L)=C(K,L)+B(K,I)*AI(I,L)                                  HM050429
C  MULT. AI*C + AR   STO. B                                         HM050430
      DO 5 K=1,KSZ                                                  HM050431
      DO 5 L=1,KSZ                                                  HM050432
      B(K,L)=AR(K,L)                                                HM050433
      DO 5 I=1,KSZ                                                  HM050434
 5    B(K,L)=B(K,L)+AI(K,I)*C(I,L)                                  HM050435
      NO=0                                                          HM050436
      CALL INVERS(B,KSZ,NO)                                         HM050437
      IF (NO) 2,7,2                                                 HM050438
C  SECOND MATRIX  NOT SINGULAR                                      HM050439
C  MULT. -C*B  STO. AI      ALSO SET AR=B                           HM050440
 7    DO 8 K=1,KSZ                                                  HM050441
      DO 8 L=1,KSZ                                                  HM050442
      AI(K,L)=0.0                                                   HM050443
      AR(K,L)=B(K,L)                                                HM050444
      DO 8 I=1,KSZ                                                  HM050445
```

```
 8    AI(K,L)=AI(K,L)-C(K,I)*B(I,L)                                    HM050446
      GO TO 20                                                         HM050447
C  REAL MATRIX OR SECOND MATRIX SINGULAR     TRY IMAG. ROUTE           HM050448
 2    DO 9 K=1,KSZ                                                     HM050449
      DO 9 L=1,KSZ                                                     HM050450
 9    B(K,L)=AI(K,L)                                                   HM050451
      NO=0                                                             HM050452
      CALL INVERS(B,KSZ,NO)                                            HM050453
      IF (NO) 10,11,10                                                 HM050454
C  IMAG. NOT SINGULAR                                                  HM050455
C  MULT. B*AR   STO. C                                                 HM050456
 11   DO 12 K=1,KSZ                                                    HM050457
      DO 12 L=1,KSZ                                                    HM050458
      C(K,L)=0.0                                                       HM050459
      DO 12 I=1,KSZ                                                    HM050460
 12   C(K,L)=C(K,L)+B(K,I)*AR(I,L)                                     HM050461
C  MULT. AR*C+AI   STO B                                               HM050462
      DO 13 K=1,KSZ                                                    HM050463
      DO 13 L=1,KSZ                                                    HM050464
      B(K,L)=AI(K,L)                                                   HM050465
      DO 13 I=1,KSZ                                                    HM050466
 13   B(K,L)=B(K,L)+AR(K,I)*C(I,L)                                     HM050467
      NO=0                                                             HM050468
      CALL INVERS(B,KSZ,NO)                                            HM050469
      IF (NO) 10,15,10                                                 HM050470
C  THIRD MATRIX NOT SINGULAR                                           HM050471
C  MULT -C*B   STO AR    ALSO SET AI=-B                                HM050472
 15   DO 16 K=1,KSZ                                                    HM050473
      DO 16 L=1,KSZ                                                    HM050474
      AR(K,L)=0                                                        HM050475
      AI(K,L)=-B(K,L)                                                  HM050476
      DO 16 I=1,KSZ                                                    HM050477
 16   AR(K,L)=AR(K,L)+C(K,I)*B(I,L)                                    HM050478
      GO TO 20                                                         HM050479
 10   I GOOF D=1                                                       HM050480
 20   RETURN                                                           HM050481
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

```
                                        STORAGE NOT USED BY PROGRAM

        DEC   OCT               DEC   OCT
        516 01004             32561 77461

   STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

        DEC   OCT               DEC   OCT               DEC   OCT               DEC   OCT
   NC   515 01003

                          LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

        DEC   OCT               DEC   OCT               DEC   OCT               DEC   OCT
    2)  501 00765         3)    504 00770         6)    505 00771         9)    511 00777
 C)100  513 01001      C)101    514 01002      D)20E    230 00346      D)40L    286 00436
 D)411  364 00554      D)41E    465 00721

                                     LOCATIONS OF NAMES IN TRANSFER VECTOR

        DEC   OCT               DEC   OCT               DEC   OCT               DEC   OCT
INVERS    0 00000

                               ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

INVERS

             EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

  EFN   IFN   LOC          EFN   IFN   LOC          EFN   IFN   LOC          EFN   IFN   LOC
  101     6 00204          102     9 00211            1    12 00224            3    17 00245
    4    21 00276            5    26 00354            7    31 00410            8    36 00446
    2    38 00472            9    40 00505           11    45 00526           12    49 00562
   13    54 00637           15    59 00673           16    64 00734           10    66 00757
   20    67 00761
```

```
      SUBROUTINE INVERS (A,N,I GOOF O)                                  HM050483
      DIMENSION A(6,6), L(6), M(6)                                      HM050484
      IF ACCUMULATOR OVERFLOW 500,500                                   HM050485
  500 IF QUOTIENT OVERFLOW 501,501                                      HM050486
  501 IF DIVIDE CHECK 502,502                                           HM050487
  502 IGOOFD=0                                                          HM050488
C     SEARCH FOR LARGEST ELEMENT                                        HM050489
      DO 80 K=1,N                                                       HM050490
      L(K)=K                                                            HM050491
      M(K)=K                                                            HM050492
      BIGA=A(K,K)                                                       HM050493
      DO 20 I=K,N                                                       HM050494
      DO 20 J=K,N                                                       HM050495
      IF(ABSF(BIGA)-ABSF(A(I,J)))10,20,20                               HM050496
   10 BIGA=A(I,J)                                                       HM050497
      L(K)=I                                                            HM050498
      M(K)=J                                                            HM050499
   20 CONTINUE                                                          HM050500
C     INTERCHANGE ROWS                                                  HM050501
      JROW=L(K)                                                         HM050502
      IF(L(K)-K)35,35,25                                                HM050503
   25 DO 30 I=1,N                                                       HM050504
      HOLD=-A(K,I)                                                      HM050505
      A(K,I)=A(JROW,I)                                                  HM050506
   30 A(JROW,I)=HOLD                                                    HM050507
C     INTERCHANGE COLUMNS                                               HM050508
   35 ICOL=M(K)                                                         HM050509
      IF(M(K)-K)45,45,37                                                HM050510
   37 DO 40 J=1,N                                                       HM050511
      HOLD=-A(J,K)                                                      HM050512
      A(J,K)=A(J,ICOL)                                                  HM050513
   40 A(J,ICOL)=HOLD                                                    HM050514
C     DIVIDE COLUMN BY MINUS PIVOT                                      HM050515
   45 DO 55 IC=1,N                                                      HM050516
   46 IF(IC-K)50,55,50                                                  HM050517
   50 A(IC,K)=A(IC,K)/(-A(K,K))                                         HM050518
   55 CONTINUE                                                          HM050519
C     REDUCE MATRIX                                                     HM050520
```

```
      DO 65 I=1,N                                                       HM050521
      DO 65 J=1,N                                                       HM050522
   56 IF(I-K)57,65,57                                                   HM050523
   57 IF(J-K)60,65,60                                                   HM050524
   60 A(I,J)=A(I,K)*A(K,J)+A(I,J)                                       HM050525
   65 CONTINUE                                                          HM050526
C     DIVIDE ROW BY PIVOT                                               HM050527
      DO 75 JR=1,N                                                      HM050528
   68 IF(JR-K)70,75,70                                                  HM050529
   70 A(K,JR)=A(K,JR)/A(K,K)                                            HM050530
   75 CONTINUE                                                          HM050531
C     CONTINUED PRODUCT OF PIVOTS                                       HM050532
C     REPLACE PIVOT BY RECIPROCAL                                       HM050533
      A(K,K)=1.0/A(K,K)                                                 HM050534
C     CONTINUE COMPLETE OPERATION                                       HM050535
   80 CONTINUE                                                          HM050536
      IF DIVIDE CHECK510,503                                            HM050537
  503 IF ACCUMULATOR OVERFLOW 510,504                                   HM050538
  504 IF QUOTIENT OVERFLOW 510,505                                      HM050539
C     FINAL ROW AND COLUMN INTERCHANGE                                  HM050540
  505 K=N                                                               HM050541
  100 K=(K-1)                                                           HM050542
      IF(K)150,150,103                                                  HM050543
  103 I=L(K)                                                            HM050544
      IF(I-K)120,120,105                                                HM050545
  105 DO 110 J=1,N                                                      HM050546
      HOLD=A(J,K)                                                       HM050547
      A(J,K)=-A(J,I)                                                    HM050548
  110 A(J,I)=HOLD                                                       HM050549
  120 J=M(K)                                                            HM050550
      IF(J-K)100,100,125                                                HM050551
  125 DO 130 I=1,N                                                      HM050552
      HOLD=A(K,I)                                                       HM050553
      A(K,I)=-A(J,I)                                                    HM050554
  130 A(J,I)=HOLD                                                       HM050555
      GO TO 100                                                         HM050556
  150 RETURN                                                            HM050557
  510 I GOOF D=1                                                        HM050558
```

HM050559

```
      GO TO 150
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
516	01004	32561	77461

STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
L	515	01003	M	509	00775						

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
BIGA	503	00767	HOLD	502	00766	ICOL	501	00765	IC	500	00764
I	499	00763	JROW	498	00762	JR	497	00761	J	496	00760
K	495	00757									

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
1)	475	00733	2)	464	00720	3)	467	00723	4)	32767	77777
6)	468	00724	9)	474	00732	C)GO	479	00737	C)100	480	00740
C)101	481	00741	C)102	482	00742	C)104	483	00743	C)106	484	00744
C)107	485	00745	C)108	486	00746	C)109	487	00747	C)10A	488	00750
C)20C	489	00751	C)202	490	00752	C)203	491	00753	C)205	492	00754
C)206	493	00755	C)207	494	00756	D)10C	194	00302	D)10E	207	00317
D)10J	246	00366	D)10P	282	00432	D)10R	292	00444	D)119	415	00637
D)11B	426	00652	D)11D	444	00674	D)11F	456	00710	D)20N	273	00421
D)20U	322	00502	D)30J	245	00365	D)30U	321	00501	D)319	414	00636
D)31D	443	00673	D)40G	224	00340	D)40I	237	00355	D)70J	244	00364
D)719	413	00635	D)71D	442	00672	E)A	185	00271	E)T	311	00467
E)17	381	00575									

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
500	6	00106	501	8	00111	502	9	00113	10	17	00250
20	20	00256	25	23	00301	30	26	00307	35	27	00320
37	29	00327	40	32	00345	45	33	00356	46	34	00367
50	35	00372	55	36	00377	56	39	00427	57	40	00433
60	41	00436	65	42	00445	68	44	00503	70	45	00506
75	46	00511	80	48	00521	503	50	00554	504	52	00557
505	54	00562	100	55	00565	103	57	00576	105	59	00605
110	62	00644	120	63	00653	125	65	00662	130	68	00701
150	70	00711	510	72	00715						

```
                       ***********************************************************************HM051203
                       *                                                                      HM051204
                       * CALLING SEQUENCE                                                     HM051205
                       *       TSX      BINPU,4                                               HM051206
                       *       TSX      LOC (ARRAY TO BE PUNCHED)                             HM051207
                       *       TSX      LOC (NO. WORDS TO PUNCH)                              HM051208
                       *       TSX      LOC (CARD ORIGIN FOR 1ST CARD)                        HM051209
                       *      *TSX      LOC (SEQ NO. OF 1ST CARD)                             HM051210
                       *      *TSX      LOC (BCD ID FOR THIS DECK, 1ST AND 2ND CHARACTER BLANKS) HM051211
                       *       TSX      LOC (OUTPUT TAPE NUMBER)                              HM051212
                       *****CONTRARY TO BELOW, NO ITEMS MAY BE OMITTED IN THIS MODIFICATION.  HM051213
                       *                                                                      HM051214
                       * ITEMS MARKED (*) MAY BE DELETED. BCD ID WILL BE                      HM051215
                       *      UNCHANGED AND SEQ. NOS. WILL BE CONTINUOUS STARTING             HM051216
                       *      FROM  000. ALSO ORDER MAY BE SWITCHED.                          HM051217
                       *                                                                      HM051218
                       * THIS VERSION PUNCHES OFF-LINE ONLY.                                  HM051219
                       *                                                                      HM051220
                       ***********************************************************************HM051221
               00006          ENTRY   BINPU                                                   HM051222
                       *                                                                      HM051223

 TRANSFER VECTOR
00000  743146623460     (IOS)
00001  746651623460     (WRS)
00002  745123303460     (RCH)
00003  746663233460     (WTC)
00004  746625513460     (WER)
00005  746325623460     (TES)

00006  0634 00 1 00142    BINPU SXA    X1,1                                                   HM051224
00007  0634 00 2 00143          SXA    X2,2                                                   HM051225
00010 -0500 60 4 00006          CAL*   6,4                                                    HM051226
00011  0622 00 0 00331          STD    14D                                                    HM051227
00012  0500 00 4 00001          CLA    1,4                 LOC OF ARRAY                       HM051228
00013  0621 00 0 00062          STA    ARRAY                                                  HM051229
00014 -0500 60 4 00002          CAL*   2,4                 WORD COUNT                         HM051230
00015  0602 00 0 77776          SLW    END                 END=0 IF TRANSFER CARD             HM051231
```

```
00016  0402 00 0 00325        SUB      D1                                                  HM051232
00017  0622 00 0 00066        STD      LOCN                                                HM051233
00020  0634 00 0 00061        SXA      COUNT,0                                             HM051234
00021  0500 60 4 00003        CLA*     3,4                 SET UP CONTROL WORD             HM051235
00022  0771 00 0 00022        ARS      18                                                  HM051236
00023 -0120 00 0 00025        TMI      *+2                                                 HM051237
00024 -0501 00 0 00266        ORA      REL                 ADD RELATIVE BIT                HM051238
00025 -0501 00 0 00334        ORA      IMAGE               7-9, WORD COUNT=22              HM051239
00026  0602 00 0 77740        SLW      CIMAGE              CONTROL WORD ESTABLISHED.       HM051240
                         *****************************************************************HM051241
                         *        TEST FOR FOURTH AND-OR FIFTH ARGUMENTS.                  HM051242
                         *        DETERMINE WHETHER ARGUMENT REFERS TO ID OR SEQ NUMBER    HM051243
                         *        AND SET CELLS FROM CALLING SEQUENCE.                     HM051244
                         *****************************************************************HM051245
00027  0774 00 2 00002        AXT      2,2                                                 HM051246
00030 -0625 00 0 00302        STL      BLSEQ               SET BLSEQ TO ITS NORMAL STATE   HM051247
00031 -0500 00 4 00004   G4   CAL      4,4                 TEST FOR 4TH, 5TH ARGS          HM051248
00032 -0320 00 0 00265        ANA      MSKPDT                                              HM051249
00033  0322 00 0 00307        ERA      MSKTSX                                              HM051250
00034 -0100 00 0 00054        TNZ      G2                  NO MORE TSXES                   HM051251
00035  0500 60 4 00004        CLA*     4,4                                                 HM051252
00036 -0340 00 0 00262        LAS      BCI8                                                HM051253
00037  0020 00 0 00051        TRA      G3                  BIG, THIS IS ID                 HM051254
00040  0600 00 0 00302        STZ      BLSEQ               EQUAL, FLAG BLANK SEQ. NO.      HM051255
00041 -0100 00 0 00043        TNZ      *+2                 IS SEQ NO NON-ZERO.             HM051256
00042 -0754 00 0 00000        PXD                          NO                              HM051257
00043 -0130 00 0 00000        XCL                          SMALL, THIS IS SEQ NO.          HM051258
00044  0634 00 4 00046        SXA      *+2,4                                               HM051259
00045  0074 00 4 00172        TSX      COSEQ,4             CONVERT SEQ NO TO BCD           HM051260
00046  0774 00 4 00000        AXT      **,4                                                HM051261
00047  0602 00 0 00267        SLW      SEQNO               SAVE                            HM051262
00050  1 77777 4 00053        TXI      G5,4,-1                                             HM051263
00051  0601 00 0 00305   G3   STO      BCDID                                               HM051264
00052  1 77777 4 00053        TXI      G5,4,-1             MOVE TO NEXT ARGUMENT           HM051265
00053  2 00001 2 00031   G5   TIX      G4,2,1              AT MOST 2 EXTRA ARGS.           HM051266
00054  0634 00 4 00144   G2   SXA      X4,4                                                HM051267
00055 -0520 00 0 77776        NZT      END                 IS WORD COUNT ZERO              HM051268
00056  0020 00 0 00152        TRA      TRCD                MUST BE A TRANSFER CARD         HM051269
```

```
                              *                                                                     HM051270
                              ***********************************************************************HM051271
                              *      BUILD THE CARD IMAGE.                                          HM051272
                              ***********************************************************************HM051273
                              *                                                                     HM051274
   00057  0774 00 2 00026  NEXT   AXT     22,2                                                      HM051275
T  00060 -0754 00 0 00000         PXD                          CLEAR AC FOR CHECKSUM.               HM051276
   00061  0774 00 4 00000  COUNT  AXT     **,4                                                      HM051277
   00062  0560 00 4 00000  ARRAY  LDQ     **,4                 MOVE ARRAY INTO CORE.                HM051278
   00063 -0600 00 2 77770         STQ     CIMAGE+24,2                                               HM051279
   00064  0361 00 2 77770         ACL     CIMAGE+24,2          ACCUMULATE CHECKSUM FOR BODY.        HM051280
   00065  1 00001 4 00066         TXI     *+1,4,1                                                   HM051281
   00066  3 00000 4 00160  LOCN   TXH     OUT,4,**             FINISH WHEN SPECIFIED                HM051282
   00067  2 00001 2 00062         TIX     ARRAY,2,1            BY NO. WORDS DESIRED.(2,4)           HM051283
   00070  0634 00 4 00061  IN     SXA     COUNT,4              SET COUNT FOR NEXT LOOP.             HM051284
   00071  0361 00 0 77740         ACL     CIMAGE               ADD IN CONTROL WORD.                 HM051285
   00072  0602 00 0 77741         SLW     CIMAGE+1             PUT CHECKSUM IN IMAGE.               HM051286
                              *                                                                     HM051287
                              ***********************************************************************HM051288
                              *      EDIT THE IDENTIFICATION FIELD.                                 HM051289
                              ***********************************************************************HM051290
                              *                                                                     HM051291
   00073 -0500 00 0 00267  EDIT   CAL     SEQNO                                                     HM051292
   00074  0560 00 0 00327         LDQ     L(1)                                                      HM051293
   00075 -0765 00 0 00022         LGR     18                                                        HM051294
   00076 -0500 00 0 00305         CAL     BCDID                                                     HM051295
   00077 -0763 00 0 00006         LGL     6                                                         HM051296
   00100 -0600 00 0 00326         STQ     IDLCD            SAVE FOR FINISHING.                      HM051297
   00101 -0130 00 0 00000         XCL                                                               HM051298
   00102  0634 00 1 00120         SXA     SV1,1                                                     HM051299
   00103  0774 00 2 00004         AXT     4,2                                                       HM051300
   00104  0774 00 4 00002         AXT     2,4                                                       HM051301
   00105  0774 00 1 00003  ABC    AXT     3,1                                                       HM051302
T  00106 -0754 00 0 00000         PXD                                                               HM051303
   00107 -0114 01 0 00230         CAQ     TAB,,1                                                    HM051304
   00110 -2 00001 1 00113         TNX     *+3,1,1                                                   HM051305
   00111  0767 00 0 00014         ALS     12                                                        HM051306
   00112  0020 00 0 00107         TRA     *-3                                                       HM051307
```

```
00113  0602 00 2 77734         SLW    LAST+4,2        COL BIN AT LAST TO LAST+3            HM051308
00114  1 77777 2 00115         TXI    *+1,2,-1                                             HM051309
00115  2 00001 4 00105         TIX    ABC,4,1                                              HM051310
00116  0560 00 0 00326         LDQ    IDLCD           FINISH W/SAVED C(MQ).                HM051311
00117  3 00000 2 00104         TXH    ABC-1,2,0                                            HM051312
00120  0774 00 1 00000  SV1    AXT    **,1                                                 HM051313
                        *                                                                  HM051314
                        *******************************************************************HM051315
                        *      THE ENTIRE CARD IMAGE IS BUILT, WITH THE BODY               HM051316
                        *      AT CIMAGE THRU CIMAGE+23, AND ID AT LAST THRU LAST+3.       HM051317
                        *      NOW  ***** WRITE THE CARD ON TAPE. ********                 HM051318
                        *******************************************************************HM051319
00121  0761 00 0 00000  WRITE  NOP                                                         HM051320
00122 -0500 00 0 00331  WRITE1 CAL    14D                                                  HM051321
00123  0074 00 4 00000         CALL   (IOS)           ESTABLISH I/O FOR TAPE 14.           HM051322
00124  0522 60 0 00001         XEC*   $(WRS)                                               HM051323
00125 -0774 00 4 00213         AXC    PUNCMD,4                                             HM051324
00126  0522 60 0 00002         XEC*   $(RCH)                                               HM051325
00127  0754 00 4 00000         PXA    0,4             SET (WER) FOR RETRY.                 HM051326
00130  0621 60 0 00003         STA*   $(WTC)                                               HM051327
00131  0074 00 4 00004  BPTES  TSX    $(WER),4                                             HM051328
00132 -0500 00 0 00267         CAL    SEQNO                                                HM051329
00133  0400 00 0 00327         ADD    L(1)            INCREMENT CARD COUNT.                HM051330
00134  0114 06 0 00215         CVR    TB1,,6                                               HM051331
00135  0602 00 0 00267         SLW    SEQNO                                                HM051332
00136  0520 00 0 77776         ZET    END             TEST IF LAST CARD.                   HM051333
00137  0020 00 0 00146         TRA    SWTCH               NOT THE LAST CARD....            HM051334
00140 -0500 00 0 00131         CAL    BPTES                                                HM051335
00141  0602 60 0 00005         SLW*   $(TES)                                               HM051336
00142  0774 00 1 00000   X1    AXT    **,1                ALL DONE. EXIT                   HM051337
00143  0774 00 2 00000   X2    AXT    **,2                                                 HM051338
00144  0774 00 4 00000   X4    AXT    **,4                                                 HM051339
00145  0020 00 4 00005         TRA    5,4                                                  HM051340
                        *                                                                  HM051341
                        *******************************************************************HM051342
00146 -0500 00 0 77740  SWTCH  CAL    CIMAGE              UPDATE THE CARD ORIGIN.          HM051343
00147  0361 00 0 00333         ACL    A22                                                  HM051344
00150  0602 00 0 77740         SLW    CIMAGE                                               HM051345
```

```
00151  0020 00 0 00057          TRA    NEXT                                            HM051346
        ***********************************************************************HM051347
        *                                                                      HM051348
00152  0774 00 2 00027   TRCD  AXT    23,2              CLEAR REST OF            HM051349
00153  0600 00 2 77770         STZ    CIMAGE+24,2       CARD IMAGE.              HM051350
00154  2 00001 2 00153         TIX    *-1,2,1                                    HM051351
00155  0500 00 0 00322         CLA    ZWC               STORE ZERO WORD COUNT    HM051352
00156  0622 00 0 77740         STD    CIMAGE                                     HM051353
00157  0020 00 0 00073         TRA    EDIT                                       HM051354
        *                                                                      HM051355
        ***********************************************************************HM051356
        *                                                                      HM051357
00160  0600 00 0 77776   OUT   STZ    END               WORD COUNT EXHAUSTED     HM051358
00161 -2 00001 2 00070         TNX    IN,2,1            RETURN IF CARD IS FULL   HM051359
00162  0602 00 0 77777         SLW    COMMON            SAVE CHECKSUM.           HM051360
00163 -0754 00 2 00000         PXD    0,2               CORRECT WORD COUNT       HM051361
00164  0402 00 0 77740         SUB    CIMAGE                                     HM051362
00165  0622 00 0 77740         STD    CIMAGE                                     HM051363
00166 -0500 00 0 77777         CAL    COMMON            RETURN CHECKSUM.         HM051364
00167 -3 00000 2 00070         TXL    IN,2,0                                     HM051365
00170  0600 00 2 77770         STZ    CIMAGE+24,2       CLEAR REST OF CARD.      HM051366
00171  1 77777 2 00167         TXI    *-2,2,-1                                   HM051367
        *                                                                      HM051368
        ***********************************************************************HM051369
        * THIS ROUTINE CONVERTS A BINARY INTEGER TO BCD. (4 DIGITS DECR-MQ)    HM051370
        ***********************************************************************HM051371
        *                                                                      HM051372
T  00172 -0754 00 0 00000  COSEQ PXD                                          HM051373
   00173 -0520 00 0 00302        NZT    BLSEQ        TEST IF BLANKS DESIRED.   HM051374
   00174  0020 00 0 00211        TRA    COSEQX                                 HM051375
   00175  0765 00 0 00022        LRS    18           RIGHT ADJUST BIN INTEGER  HM051376
   00176  0221 00 0 00332        DVP    TEN                                    HM051377
   00177  0601 00 0 77777        STO    COMMON                                 HM051378
T  00200 -0754 00 0 00000        PXD                                           HM051379
   00201  0221 00 0 00332        DVP    TEN                                    HM051380
   00202  0767 00 0 00006        ALS    6                                      HM051381
   00203 -0602 00 0 77777        ORS    COMMON                                 HM051382
T  00204 -0754 00 0 00000        PXD                                           HM051383
```

```
00205  0221 00 0 00332         DVP    TEN                                              HM051384
00206  0767 00 0 00014         ALS    12                                               HM051385
00207 -0501 00 0 77777         ORA    COMMON         RIGHT ADJUSTED SEQ NO.            HM051386
00210  0020 00 4 00001         TRA    1,4                                              HM051387
00211 -0500 00 0 00306  COSEQX CAL    BLANK          BLANK OUT SEQ. NO.                HM051388
00212  0020 00 4 00001         TRA    1,4                                              HM051389
                        *                                                              HM051390
                        ***************************************************************HM051391
                        *                                                              HM051392
00213 -0 00030 0 77740  PUNCMD IOCP   CIMAGE,0,24                                      HM051393
00214  0 00003 0 77730         IOCD   LAST,0,3                                         HM051394
                        *                                                              HM051395
                        * TABLE FOR BCD ADDITION OF 1 TO C(ACC)                        HM051396
                        *                                                              HM051397
00215  0000 00 0 00215  TB1    HTR    TB1               0                              HM051398
00216  0100 00 0 00215  TB     TZE    TB1               1                              HM051399
00217  0200 00 0 00215         MPY    TB1               2                              HM051400
00220  0300 00 0 00215         FAD    TB1               3                              HM051401
00221  0400 00 0 00215         ADD    TB1               4                              HM051402
00222  0500 00 0 00215         CLA    TB1               5                              HM051403
00223  0600 00 0 00215         STZ    TB1               6                              HM051404
00224  0700 00 0 00215         CPY    TB1               7                              HM051405
00225  1 00000 0 00215         TXI    TB1               8                              HM051406
00226  1 10000 0 00215         TXI    TB1,0,4096        9                              HM051407
00227  0000 00 0 00216         HTR    TB                0 WITH CARRY                   HM051408
                        * TABLES FOR BCD-COL. BIN. CONVERSION                          HM051409
                        * HOLES ARE FILLED IN WITH CONSTANTS                           HM051410
00230 +000000001000     TAB    OCT    1000,400,200,100,40,20,10,4,2,1                  HM051411
00231 +000000000400
00232 +000000000200
00233 +000000000100
00234 +000000000040
00235 +000000000020
00236 +000000000010
00237 +000000000004
00240 +000000000002
00241 +000000000001
00242 -377777770000     MSK2CH OCT    777777770000,102,42                              HM051412
```

4
TD

```
00243 +000000000102
00244 +000000000042
00245 +000000000000      ID123 OCT     0,0,0                                                      HM051413
00246 +000000000000
00247 +000000000000
00250 +000000004000            OCT     4000,4400,4200,4100,4040,4020,4010,4004,4002,4001          HM051414
00251 +000000004400
00252 +000000004200
00253 +000000004100
00254 +000000004040
00255 +000000004020
00256 +000000004010
00257 +000000004004
00260 +000000004002
00261 +000000004001
00262  106060606060      BCI8  BCI     1,8                                                        HM051415
00263 +000000004102            OCT     4102,4042                                                  HM051416
00264 +000000004042
00265 -3 77777 7 00000  MSKPDT TXL     0,7,-1                                                     HM051417
00266  0400 00 0 00000   REL   ADD                                                                HM051418
00267  606060606060      SEQNO BCI     1,                                                         HM051419
00270 +000000002000            OCT     2000,2400,2200,2100,2040,2020,2010,2004,2002,2001          HM051420
00271 +000000002400
00272 +000000002200
00273 +000000002100
00274 +000000002040
00275 +000000002020
00276 +000000002010
00277 +000000002004
00300 +000000002002
00301 +000000002001
00302  0 00000 0 00000   BLSEQ                                                                    HM051421
00303 +000000002102            OCT     2102,2042                                                  HM051422
00304 +000000002042
00305  606060606060      BCDID BCI     1,                                                         HM051423
00306  606060606060      BLANK BCI     1,                                                         HM051424
00307  0074 00 0 00000  MSKTSX TSX     ,0                                                         HM051425
00310 +000000000000            OCT     0,1400,1200,1100,1040,1020,1010,1004,1002,1001             HM051426
```

```
00311 +000000001400
00312 +000000001200
00313 +000000001100
00314 +000000001040
00315 +000000001020
00316 +000000001010
00317 +000000001004
00320 +000000001002
00321 +000000001001
00322 +000500000000      ZWC    OCT      000500000000        ZERO WORDS FOR TCD               HM051427
00323 +000000001102             OCT      1102,1042                                            HM051428
00324 +000000001042
00325  0 00001 0 00000    D1             0,0,1                                                HM051429
00326  0 00000 0 00000  IDLCD   PZE                                                           HM051430
00327  0 00000 0 00001  L(1)    PZE      1                                                    HM051431
00330  0 00000 0 00005  5A      PZE      5                                                    HM051432
00331  0 00014 0 00020   14D    PZE 16,,12                                                    HM051433
00332 +000000000012     TEN     DEC      10                                                   HM051434
00333  0000 00 0 00026   A22    HTR      22                                                   HM051435
00334 +000526000000      IMAGE  OCT      000526000000        CONTROL WORD SKELETON            HM051436
                 77777  COMMON  EQU      -1                                                   HM051437
                 77740  CIMAGE  EQU      -32                                                  HM051438
                 77730  LAST    EQU      -40                                                  HM051439
                 77776   END    SYN      COMMON-1                                             HM051440
                                END                                                           HM051441
```

```
BINPU ROUTINE TO WRITE COL BIN CARDS ON TAPE. FIBII
              POST PROCESSOR ASSEMBLY DATA

335 IS THE FIRST LOCATION NOT USED BY THIS PROGRAM


REFERENCES TO DEFINED SYMBOLS
        330      5A
        325      D1     16
         54      G2     34
         51      G3     37
         31      G4     53
         53      G5     50,   52
         70      IN    161,  167
        216      TB    227
        142      X1      6
        143      X2      7
        144      X4     54
        331     14D     11,  122
        333     A22    147
        105     ABC    115,  117
      77776     END     15,   55,  136,  160,  335
        160     OUT     66
        266     REL     24
        120     SV1    102
        230     TAB    107
        215     TB1    134,  215,  216,  217,  220,  221,  222,  223,  224,  225,  226
        332     TEN    176,  201,  205
        322     ZWC    155
        262    BCI8     36
         73    EDIT    157
      77730    LAST    113,  214,  335
         66    LOCN     17
        327    L(1)     74,  133
         57    NEXT    151
        152    TRCD     56
         62   ARRAY     13,   67
        305   BCDID     51,   76
          6   BINPU
        306   BLANK    211
```

POST PROCESSOR ASSEMBLY DATA

```
  302  BLSEQ    30,   40,  173
  131  BPTES   140
  172  COSEQ    45
   61  COUNT    20,   70
  245  ID123
  326  IDLCD   100,  116
  334  IMAGE    25
  267  SEQNO    47,   73,  132,  135
  146  SWTCH   137
  121  WRITE
    0  (IOS)   123
    2  (RCH)   126
    5  (TES)   141
    4  (WER)   131
    1  (WRS)   124
    3  (WTC)   130
77740 CIMAGE    26,   63,   64,   71,   72,  146,  150,  153,  156,  164,  165,  170,  213,  335
77777 COMMON   162,  166,  177,  203,  207,  335
  211 COSEQX   174
  242 MSK2CH
  265 MSKPDT    32
  307 MSKTSX    33
  213 PUNCMD   125
  122 WRITE1
```

NO ERROR IN ABOVE ASSEMBLY.

```
C  MATRIX COLUMN BINARY PUNCH SUBROUTINE                                HM051443
C  CALLING SEQUENCE.........                                            HM051444
C    CALL MPUNCH (A,M,N,IDUT,ITRA,IORG,BCOZ,MAXM,NTAPE   )              HM051445
C                                                                       HM051446
C              A = MATRIX TO BE PUNCHED      IORG = ORIGIN OF FIRST CARD  HM051447
C              M = NUMBER OF ROWS            BCOZ = BCD IDENTIFICATION WRD HM051448
C              N = NUMBER OF COLUMNS         MAXM = DIMENSIONED NUMBER OF  HM051449
C           IDUT = 0, PUNCH BY COLUMNS                          ROWS     HM051450
C                = 1, PUNCH BY ROWS        NTAPE= OUTPUT PUNCH TAPE      HM051451
C           ITRA = 0, TRA CARD AFTER WHOLE MATRIX                       HM051452
C                = 1, TRA CARD AFTER EACH ROW OR COLUMN                 HM051453
                                                                        HM051454
      SUBROUTINE MPUNCH (A,M,N,IDUT,ITRA,IORG,BCDZ,MAXM,NTAPE,NCARDS )  HM051455
                                                                        HM051456
      DIMENSION   A(1), T(22)                                           HM051457
                                                                        HM051458
      IS=NCARDS                                                         HM051459
      MN = MAXM*N                                                       HM051460
      IF  ( IDUT )   8,2,8                                              HM051461
                                                                        HM051462
                                                                        HM051463
C PUNCH BY COLUMNS                                                      HM051464
                                                                        HM051465
    2 J = 1                                                             HM051466
      J1= IORG                                                          HM051467
      DO  5  I=1,N                                                      HM051468
      CALL BINPU  (A(J), M, J1, BCDZ, IS, NTAPE )                       HM051469
      J = J+MAXM                                                        HM051470
      IS= IS+1+M/22                                                     HM051471
      IF  ( ITRA )  3,4,3                                               HM051472
                                                                        HM051473
C PUNCH TRA CARD AFTER EACH COLUMN                                      HM051474
                                                                        HM051475
    3 CALL BINPU  (A, 0, 0, BCDZ, IS, NTAPE )                           HM051476
      IS=IS+1                                                           HM051477
      GOTO   5                                                          HM051478
                                                                        HM051479
```

```
    4 J1=J1+M
    5 CONTINUE

      IF  ( ITRA )  16,6,16

C PUNCH TRA CARD AFTER WHOLE MATRIX

    6 CALL BINPU  (A, 0, 0, BCDZ, IS, NTAPE)
      GOTO  16


C PUNCH MATRIX BY ROWS.
    8 J=0
      J1=IORG
      DO  14  I=1,M
      DO  10  K=I,MN,MAXM
        J=J+1
        T(J)=A(K)
        IF (J-22 )   10,9,9
    9 CALL BINPU  (T,22,J1,BCDZ,IS,NTAPE)
      J1=J1+22
      IS=IS+1
      J=0
   10 CONTINUE

      IF  ( J )  12,12,11
   11 CALL BINPU  (T,J,J1,BCDZ,IS,NTAPE )
      J1=J1+J
      IS=IS+1
   12 IF ( ITRA )  13,14,13

C PUNCH TRA CARD AFTER EVERY ROW
   13 CALL BINPU  (T,0,0,BCDZ,IS,NTAPE )
      IS=IS+1
      J1=IORG
   14 J=0

C OR AFTER ENTIRE MATRIX
```

```
   IF  ( ITRA )  16,15,16                                    HM051518
15 CALL BINPU  (A,0,0,BCDZ,IS,NTAPE)                         HM051519
   IS=IS+1                                                   HM051520
16 NCARDS=IS-1                                               HM051521
   RETURN                                                    HM051522
                                                             HM051523
   END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

```
                                STORAGE NOT USED BY PROGRAM

       DEC   OCT                DEC   OCT
       288 00440              32561 77461

                    STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

       DEC   OCT                DEC   OCT               DEC   OCT               DEC   OCT
  T    287 00437

  STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

       DEC   OCT                DEC   OCT               DEC   OCT               DEC   OCT
  I    265 00411        IS      264 00410       J1      263 00407       J       262 00406
  MN   261 00405

                          LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

       DEC   OCT                DEC   OCT               DEC   OCT               DEC   OCT
  1)   258 00402        2)      247 00367       6)      250 00372       9)      256 00400
 C)GO  260 00404        E)4     117 00165       E)6     123 00173       E)H     218 00332

                                LOCATIONS OF NAMES IN TRANSFER VECTOR

       DEC   OCT                DEC   OCT               DEC   OCT               DEC   OCT
 BINPU   0 00000

                            ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

BINPU

                EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

 EFN   IFN  LOC         EFN   IFN  LOC          EFN   IFN  LOC          EFN   IFN  LOC
   2     7 00110          3    15 00152           4    19 00166           5    20 00171
   6    22 00200          8    25 00210           9    32 00244          10    37 00266
  11    39 00274         12    43 00313          13    44 00315          14    48 00333
  15    50 00346         16    53 00360
```

```
                                                                        MTRS0340
C CALLING SEQUENCE.....                                                 MTRS0341
                                                                        MTRS0342
C  A = MATRIX, DIMENSIONED  (MAXR X 2*N) - REAL                         MTRS0343
C                           (MAXR X 4*N) - COMPLEX                      MTRS0344
C  NTAPE=UTILITY TAPE, IF + MATRIX IS IN CORE AT A, IF - ITS ON NTAPE   MTRS0345
C  N = ORDER OF MATRIX                                                  MTRS0346
C  GUESS=1ST. GUESS VECTOR, DIMENSIONED  (MAXR X 1) - REAL              MTRS0347
C                                        (MAXR X 2) - COMPLEX           MTRS0348
C  NGUESS=0, ROUTINE SUPPLIES GUESS VECTOR                              MTRS0349
C        =+1, GUESS CONTAINS GUESS VECTOR                               MTRS0350
C  NMODE=NUMBER OF EIGEN SOLUTIONS REQUESTED.                           MTRS0351
C  VECTOR=EIGENVECTORS, DIMENSIONED  (MAXR X NMODE) - REAL              MTRS0352
C                                    (MAXR X 2*NMODE) - COMPLEX         MTRS0353
C  EIGVAL=EIGENVALUES                (NMODE X 1) - REAL                 MTRS0354
C                                    (NMODE*2 X 1) - COMPLEX            MTRS0355
C  NITER=NUMBER OF ITERATIONS PER MODE                                  MTRS0356
C  NITRSP, NITROP = MAXIMUM NUMBER OF SINGLE AND DOUBLE PREC. ITERATIONMTRS0357
C  EPSP, EPOP=CONVERGENCE CRITERIA FOR SINGLE AND DOUBLE ROOTS.         MTRS0358
C  IR = ERROR INDICATOR                                                 MTRS0359
C  US=CHECK EIGENVECTORS, DIMENSIONED  (MAXR X NMODE) - REAL            MTRS0360
C                                      (MAXR X 2*NMODE) - COMPLEX       MTRS0361
C  H=WORKING AREA OF CORE   DIMENSIONED   (MAXR X (NMODE+4) - REAL      MTRS0362
C                                         (MAXR X 2*(NMODE+4) - COMPLEXMTRS0363
C              WILL CONTAIN CHECK EIGENVALUES, IF REQUESTED.            MTRS0364
C  NTAPE1=TAPE NUMBER OF OUTPUT PRINT TAPE                              MTRS0365
C              IF = 0, NO RESULTS WILL BE PRINTED                       MTRS0366
C  MAXR = DIMENSIONED NUMBER OF ROWS                                    MTRS0367
C  NC = 1, PROBLEM REAL                                                 MTRS0368
C     = 2, PROBLEM COMPLEX                                              MTRS0369
C  AITKEN = AITKEN COVERGENCE CRITERIA                                  MTRS0370
C  NAKSR,NAKDR = NUMBER OF TIMES AITKEN APPLIED IN EACH MODE.           MTRS0371
                                                                        MTRS0372
                                                                        MTRS0373
                                                                        MTRS0374
      SUBROUTINE  MITERS    (A, NTAPE, N, GUESS, NGUESS, NMODE, VECTOR, MTRS0375
     1                       EIGVAL, NITER, NITRSP, NITROP, EPSP, EPDP, MTRS0376
```

```
  2                        IR, US, H,MAXR, NC, AITKEN, NAKSR, NAKDR,  MTRS0377
  3                        NTAPE1  )                                  MTRS0378
                                                                      MTRS0379
      DIMENSION  A(1), GUESS(1), VECTOR(1), EIGVAL(1), NITER(1), US(1), MTRS0380
     1          H(1), NAKSR(1), NAKDR(1), BOOLT(6)                    MTRS0381
B     BOOLT(1)=606046652551                                           MTRS0382
B     BOOLT(2)=264346666060                                           MTRS0383
B     BOOLT(3)=243165312425                                           MTRS0384
B     BOOLT(4)=602330252342                                           MTRS0385
B     BOOLT(5)=314560234346                                           MTRS0386
B     BOOLT(6)=622560626422                                           MTRS0387
                                                                      MTRS0388
      IF ACCUMULATOR OVERFLOW       3,3                               MTRS0389
    3 IF OIVIDE CHECK           4,4                                   MTRS0390
                                                                      MTRS0391
C  FIND MATRIX ANO STORE ON TAPE IF NECESSARY                         MTRS0392
    4 I=NTAPE                                                         MTRS0393
      J2=MAXR*NC*N                                                    MTRS0394
      IF ( I )   5,8,6                                                MTRS0395
    5 I=-I                                                            MTRS0396
      REWIND I                                                        MTRS0397
      DO  1  J=1,N                                                    MTRS0398
    1   READ TAPE I,    (A(K),K=J,J2,MAXR)                            MTRS0399
      NTAPE = I                                                       MTRS0400
      GOTO  7                                                         MTRS0401
                                                                      MTRS0402
    6 REWIND NTAPE                                                    MTRS0403
      OO  2  J=1,N                                                    MTRS0404
    2   WRITE TAPE NTAPE,    (A(K),K=J,J2,MAXR)                       MTRS0405
    7 REWINO NTAPE                                                    MTRS0406
                                                                      MTRS0407
C DEFINE PROGRAM CONSTANTS ANO ZEROS.                                 MTRS0408
    8 MODE=0                                                          MTRS0409
      IR=0                                                            MTRS0410
      AT=AITKEN**2                                                    MTRS0411
      IF ( EPSP )   10,9,10                                           MTRS0412
    9 EPSP = .1E-08                                                   MTRS0413
   10 IF ( EPDP )  12,11,12                                           MTRS0414
```

```
   11 EPOP = .1E-08                                                      MTRS0415
                                                                         MTRS0416
   12 IF  ( NGUESS )  15,13,15                                           MTRS0417
                                                                         MTRS0418
   13   J1=MAXR*(NC-1)                                                   MTRS0419
        DO  14  I=1,N                                                    MTRS0420
          K=J1+I                                                         MTRS0421
          GUESS(K)=0.                                                    MTRS0422
   14     GUESS(I)=1.                                                    MTRS0423
                                                                         MTRS0424
   15 MODE = MODE+1                                                      MTRS0425
      NAKSR(MODE)=0                                                      MTRS0426
      NAKDR(MODE)=0                                                      MTRS0427
      IGO=1                                                              MTRS0428
      NITER(MODE)=0                                                      MTRS0429
      K1=NC*MAXR*(MODE-1)                                                MTRS0430
      K2=K1+1                                                            MTRS0431
      K3=NC*(MODE-1)+1                                                   MTRS0432
      K4=  NC*MAXR                                                       MTRS0433
      K5=K4*NMODE                                                        MTRS0434
      K6=K5+K4                                                           MTRS0435
                                                                         MTRS0436
C MOVE FIRST GUESS INTO POSITION                                         MTRS0437
      DO  16  J=1,NC                                                     MTRS0438
        J1=MAXR*(J-1)                                                    MTRS0439
        DO  16  I=1,N                                                    MTRS0440
          K=K1+J1+I                                                      MTRS0441
          L=J1+I                                                         MTRS0442
   16     H(K)=GUESS(L)                                                  MTRS0443
                                                                         MTRS0444
   17 NAK=0                                                              MTRS0445
   18 NITER(MODE)=NITER(MODE)+1                                          MTRS0446
      NAK=NAK+1                                                          MTRS0447
      INDEX=0                                                            MTRS0448
      CALL DPMLTX (A,NC, H(K2),NC, VECTOR(K2), N,N,1, MAXR,MAXR,MAXR,1)  MTRS0449
      CALL NPNRMX (VECTOR(K2), H(K2), N, EIGVAL(K3), INDEX, MAXR, NC,1)  MTRS0450
                                                                         MTRS0451
C  TEST FOR SINGLE ROOT CONVERGENCE                                      MTRS0452
```

```
      DO  23  J=1,NC                                                    MTRS0453
        J1=(J-1)*MAXR                                                   MTRS0454
        K=K1+J1                                                         MTRS0455
      GOTO  (24,19,21),NAK                                              MTRS0456
   19   L=K5+J1                                                         MTRS0457
        00  20  I=1,N                                                   MTRS0458
          L=L+1                                                         MTRS0459
          K=K+1                                                         MTRS0460
          IF  ( ABSF(H(L)-H(K)) - EPSP)           20,20,24              MTRS0461
   20     CONTINUE                                                      MTRS0462
        GOTO 100                                                        MTRS0463
   21   DO  22  I=1,N                                                   MTRS0464
          K=K+1                                                         MTRS0465
          IF  ( ABSF(US(K)-H(K)) - EPSP)          22,22,24              MTRS0466
   22     CONTINUE                                                      MTRS0467
   23   CONTINUE                                                        MTRS0468
  100 IF ACCUMULATOR OVERFLOW                    108,102                MTRS0469
  102 IF DIVIDE CHECK        104,56                                     MTRS0470
  104 NP=3                                                              MTRS0471
      GOTO  109                                                         MTRS0472
                                                                        MTRS0473
C NO CONVERGENCE, SO TEST MAXIMUM NUMBER OF ITERATIONS.                 MTRS0474
   24 IF  ( NITER(MODE)-NITRSP)                   25,46,46              MTRS0475
                                                                        MTRS0476
C NOT YET EXCEEOEO, SO TRY FOR AITKENS TIME.                            MTRS0477
   25 GOTO  (40,44,31),NAK                                              MTRS0478
                                                                        MTRS0479
                                                                        MTRS0480
C TEST FOR AITKENS CONERGENCE.                                          MTRS0481
   31 GOTO  (26,36),NC                                                  MTRS0482
                                                                        MTRS0483
   26 00  28  I=1,N                                                     MTRS0484
         J=K5+I                                                         MTRS0485
         K=K1+I                                                         MTRS0486
      IF  ( US(K)-H(J) )     27,261,27                                  MTRS0487
  261 IF  ( H(K)-US(K) )     32,28,32                                   MTRS0488
   27    IF  ( (ABSF( (H(K)-US(K)) / (US(K)-H(J)) )) - AITKEN ) 28,28,32 MTRS0489
   28    CONTINUE                                                       MTRS0490
```

```
C  ALL VECTOR ELEMENTS OK, SO APPLY AITKENS SPEEDER-UPPER.
      DO  30  I=1,N
        J=K5+I
        K=K1+I
      Q=(H(K)-2.*US(K)+H(J))
      IF  ( Q )  29,30,29
   29 H(K)=H(J)- ( (US(K)-H(J))**2 / Q)
   30 CONTINUE
      NAKSR(MODE)=NAKSR(MODE) + 1
      GOTO  17

C CONVERGENCE TEST NOT MET. RESTORE AND TRY AGAIN.
   32 DO  33  L=1,NC
        J1=(L-1)*MAXR
        DO  33  I=1,N
        J=K1+J1+I
        K=K5+J1+I
        H(K)=US(J)
   33   US(J)=H(J)
      NAK=2
      GOTO  18

C IF PROBLEM COMPLEX, REPEAT ALL ABOVE FOR COMPLEX ARITHMETIC.
   36 DO  38  I=1,N
        J=K5+I
        K=K1+I
        JJ=J+MAXR
        KK=K+MAXR
        Q =  (US(K)-H(J))**2 + (US(KK)-H(JJ))**2
      IF  ( Q )            37,361,37
  361 IF  ( (H(K)-US(K))**2 + (H(KK) - US(KK))**2 )          32,38,32
   37   IF (  ((H(K)-US(K))**2 + (H(KK) - US(KK))**2) / Q-AT)  38,38,32

   38   CONTINUE

      DO  39  I=1,N
        J=K5+I
```

```
      JJ=J+MAXR                                                     MTRS0529
      K=K1+I                                                        MTRS0530
      KK=K+MAXR                                                     MTRS0531
     Q = (H(K)-2.*US(K)+H(J))**2 + (H(KK)-2.*US(KK)+H(JJ))**2       MTRS0532
     IF  ( Q )  35,39,35                                            MTRS0533
35    X=H(K)                                                        MTRS0534
                                                                    MTRS0535
      H(K)= H(J) - ( (((US(K)-H(J))**2-(US(KK)-H(JJ))**2)*(H(K)-2.* MTRS0536
     1                 US(K)+H(J))+(2.*(US(K)-H(J))*(US(KK)-H(JJ))* MTRS0537
     2                 (H(KK)-2.*US(KK)+H(JJ) )))  / Q )            MTRS0538
      H(KK)=H(JJ)-(((2.*(US(K)-H(J))*(US(KK)-H(JJ)))*(X-2.*         MTRS0539
     1           US(K)+H(J))-((US(K)-H(J))**2-(US(KK)-H(JJ))**2)    MTRS0540
     2              *(H(KK)-2.*US(KK)+H(JJ)))  / Q )                MTRS0541
39 CONTINUE                                                         MTRS0542
   NAKSR(MODE) = NAKSR(MODE) + 1                                    MTRS0543
   GOTO  17                                                         MTRS0544
                                                                    MTRS0545
                                                                    MTRS0546
40 DO  41  J=1,NC                                                   MTRS0547
     J1=MAXR*(J-1)                                                  MTRS0548
     DO  41  I=1,N                                                  MTRS0549
       K=K1+J1+I                                                    MTRS0550
       L=K5+J1+I                                                    MTRS0551
41     H(L)=H(K)                                                    MTRS0552
   GOTO  (18,56),IGO                                                MTRS0553
                                                                    MTRS0554
44 DO  45  J=1,NC                                                   MTRS0555
     J1=MAXR*(J-1)                                                  MTRS0556
     DO  45  I=1,N                                                  MTRS0557
       K=K1+J1+I                                                    MTRS0558
45     US(K)=H(K)                                                   MTRS0559
   GOTO  18                                                         MTRS0560
                                                                    MTRS0561
C  MODE DID NOT CONVERGE IN NORMAL ITERATION, SO SETUP AND TRY CLOSE MTRS0562
C                ROOTS PROCEDURE                                    MTRS0563
46 K=K6+2*K4+1                                                      MTRS0564
   L=K+2*K4                                                         MTRS0565
   J=K3+NC                                                          MTRS0566
```

```
      IF ACCUMULATOR OVERFLOW          108,113                          MTRS0567
  113 IF OIVIDE CHECK       104,49                                      MTRS0568
   49 CALL CLOSE (A,H(K2),H(K),H(K6+1),EIGVAL(K3),EIGVAL(J), INOEX1,    MTRS0569
     1            INOEX,N,NC,NITRDP,NTRY,NAKDR(MOOE),EPOP,EPSP,AITKEN,  MTRS0570
     2            AT,MAXR,US(K2),H(L),IR)                               MTRS0571
      IF ( IR )  114,52,50                                              MTRS0572
  114 NP=5                                                              MTRS0573
      GOTO 109                                                          MTRS0574
   50   IR=0                                                            MTRS0575
      NITER(MOOE) = NITRSP+NTRY                                         MTRS0576
        GOTO  56                                                        MTRS0577
   52 OO  54  J=1,NC                                                    MTRS0578
        J1=MAXR*(J-1)+INDEX1                                            MTRS0579
      JJ=MAXR*(J-1)                                                     MTRS0580
      OO  54  I=1,N                                                     MTRS0581
        K=K1+JJ+I                                                       MTRS0582
        KK=K1+K4+JJ+I                                                   MTRS0583
        LL=K6+2*K4+JJ+I                                                 MTRS0584
        L=K6+JJ+I                                                       MTRS0585
        H(K)=H(LL)                                                      MTRS0586
        H(KK)=H(L)                                                      MTRS0587
        US(K)=A(J1)                                                     MTRS0588
   54   J1=J1+K4                                                        MTRS0589
                                                                        MTRS0590
      CALL SWEEPX (VECTOR,A,H,US,EIGVAL,MOOE,N,MAXR,NC,INDEX1,EPSP)     MTRS0591
      IF ACCUMULATOR OVERFLOW          108,116                          MTRS0592
  123 NUTS=1                                                            MTRS0593
      GOTO 107                                                          MTRS0594
  116 IF OIVIDE CHECK       123,117                                     MTRS0595
  117 J1=(NC-1)*MAXR+INOEX1                                             MTRS0596
      GUESS(J1)=0.                                                      MTRS0597
      GUESS(INDEX1)=0.                                                  MTRS0598
      NITER(MOOE)=0                                                     MTRS0599
      MODE=MOOE+1                                                       MTRS0600
      IF ( NMODE-MODE )     53,55,55                                    MTRS0601
   53 NITER(MODE)=NITRSP+NTRY                                           MTRS0602
      K1=NC*MAXR*(MODE-1)                                               MTRS0603
      OO  531  I=1,NC                                                   MTRS0604
```

```
      K= K1+MAXR*(I-1)
      DO  531  J=1,N
        L=K+J
531     VECTOR(L)=0.
    GDTD  59
 55 NAKDR(MODE)=NAKDR(MODE-1)
    NAKDR(MODE-1)=0
    NITER(MODE) = NITRSP + NTRY
    K1=NC*MAXR*(MODE-1)

 56 DO  58  J=1,NC
      J1=MAXR*(J-1)+INDEX
      DO  58  I=1,N
        K=K1+MAXR*(J-1)+I
        US(K)=A(J1)
 58     J1=J1+K4
    CALL SWEEPX (VECTOR,A,H,US,EIGVAL,MODE,N,MAXR,NC,INDEX,EPSP)
    IF ACCUMULATOR OVERFLOW          108,106
106 IF DIVIDE CHECK        124,59
124 NUTS=2
107 IF (NTAPE1)           119,118,119
118 IR=MODE
    GOTD 120
119 WRITE DUTPUT TAPE NTAPE1, 131,  BOOLT(3),BOOLT(4)
    WRITE DUTPUT TAPE NTAPE1, 134, MODE
120 DD  125  I=1,NC
      K=K1+MAXR*(I-1)
      DD  125  J=1,N
        K=K+1
125     VECTOR(K)=0.
    GDTD  (117,59),NUTS
 59 J1=(NC-1)*MAXR+INDEX
    GUESS(J1)=0.
    GUESS(INDEX)=0.


 62 IF  (NMODE-MODE)      70,70,15

```

```
108 NP=1                                                            MTRS0643
109 IR=MODE                                                         MTRS0644
    IF ( NTAPE1 )          121, 70,121                              MTRS0645
121 WRITE OUTPUT TAPE NTAPE1, 131, BOOLT(NP) ,BOOLT(NP+1)           MTRS0646
    MODE=MODE-1                                                     MTRS0647
    WRITE OUTPUT TAPE NTAPE1, 132, MODE                             MTRS0648
    NMODE = MODE                                                    MTRS0649
    GOTO 70                                                         MTRS0650
                                                                    MTRS0651
 70 IF  ( NTAPE )  71,75,71                                         MTRS0652
 71 DO  73   J=1,N                                                  MTRS0653
 73   READ TAPE NTAPE,  (A(I),I=J,J2,MAXR)                          MTRS0654
    CALL DPMLTX (A,NC,VECTOR,NC,US,N,N,MODE,MAXR,MAXR,MAXR,1)       MTRS0655
    J=1                                                             MTRS0656
    K=1                                                             MTRS0657
    DO  72  I=1,MODE                                                MTRS0658
    INDEX=0                                                         MTRS0659
      CALL NPNRMX  (US(J),US(J),N,H(K),INDEX,MAXR,NC,1)             MTRS0660
    J=J+K4                                                          MTRS0661
 72 K=K+NC                                                          MTRS0662
 75 IF  ( NTAPE1 )         92,92,80                                 MTRS0663
                                                                    MTRS0664
                                                                    MTRS0665
 80 WRITE OUTPUT TAPE NTAPE1, 95                                    MTRS0666
    DO  86  I=1,MODE                                                MTRS0667
      LITR=NITER(I)-NITRSP                                          MTRS0668
      IF ( LITR )          81,82,82                                 MTRS0669
 81   LITR=0                                                        MTRS0670
    GOTO  85                                                        MTRS0671
 82 NITER(I)=NITRSP                                                 MTRS0672
    IF  (LITR-NITRDP)      85,85,87                                 MTRS0673
 87 WRITE OUTPUT TAPE NTAPE1, 94, I                                 MTRS0674
 85 GOTO  (83,84),NC                                                MTRS0675
 83   WRITE OUTPUT TAPE NTAPE1, 97, ( I,EIGVAL(I),NITER(I),LITR,    MTRS0676
  1                                NAKSR(I),NAKDR(I))               MTRS0677
      GOTO   86                                                     MTRS0678
 84   L=2*I-1                                                       MTRS0679
      WRITE OUTPUT TAPE NTAPE1, 96, ( I,EIGVAL(L),EIGVAL(L+1),NITER(I)MTRS0680
```

```
  1                                 ,LITR,NAKSR(I),NAKDR(I))         MTRS0681
86   CONTINUE                                                        MTRS0682
   IF  ( MODE )          92,92,88                                    MTRS0683
88 WRITE OUTPUT TAPE NTAPE1,    98                                   MTRS0684
   L=MODE*NC                                                         MTRS0685
   CALL MPRINT (VECTOR,N,L,MAXR,NTAPE1)                              MTRS0686
   IF  ( NTAPE )         92,92,90                                    MTRS0687
90 WRITE OUTPUT TAPE NTAPE1,    99                                   MTRS0688
   WRITE OUTPUT TAPE NTAPE1,  93, (H(I),I=1,L)                       MTRS0689
   CALL MPRINT (US,N,L,MAXR,NTAPE1)                                  MTRS0690
92 RETURN                                                            MTRS0691
93 FORMAT  ( 1H  6E16.8 )                                            MTRS0692
94 FORMAT (5H MODE  1I4, 40H HAS NOT CONVERGED IN MAXIMUM ITERATIONS)MTRS0693
95 FORMAT  (1H  5X, 6H MODE   13X,  11H EIGENVALUE   19X,            MTRS0694
  2         24HITERATIONS  S.P.    D.P.  6X, 13H AITKENS S.P.  2X,   MTRS0695
  3         5H D.P.  ///)                                            MTRS0696
96 FORMAT  (1H  1I11, 2E19.8,  1I22, 1I7, 1I19, 1I7)                 MTRS0697
97 FORMAT  (1H  1I11, 9X,  1E20.8, 9X, 1I22, 1I7, 1I19, 1I7)         MTRS0698
98 FORMAT  (1H0  /// 1H0  46X,  14H EIGENVECTORS   /// )             MTRS0699
99 FORMAT  (1H1  /// 1H0  36H CHECK EIGENVALUES AND EIGENVECTORS   ) MTRS0700
131 FORMAT (35H1ERROR IN ITERATION SUBROUTINE....(   2A6,  1H)  )    MTRS0701
132 FORMAT (25H+ CALCULATION TERMINATED.  1I6,19H MODES ARE CORRECT.) MTRS0702
134 FORMAT (14H+ IN TRUE MODE  1I6, 27H CALCULATION. MODIFIED MODE   MTRS0703
  1         12H IS CORRECT.  )                                       MTRS0704
   END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
2305	04401	32561	77461

STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

	OEC	OCT		OEC	OCT		OEC	OCT		DEC	OCT
BOOLT	2304	04400									

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, OIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		OEC	OCT		OEC	OCT
AT	2298	04372	IGO	2297	04371	INDEX1	2296	04370	INDEX	2295	04367
I	2294	04366	J1	2293	04365	J2	2292	04364	JJ	2291	04363
J	2290	04362	K1	2289	04361	K2	2288	04360	K3	2287	04357
K4	2286	04356	K5	2285	04355	K6	2284	04354	KK	2283	04353
K	2282	04352	LITR	2281	04351	LL	2280	04350	L	2279	04347
MOOE	2278	04346	NAK	2277	04345	NP	2276	04344	NTRY	2275	04343
NUTS	2274	04342	Q	2273	04341	X	2272	04340			

SYMBOLS AND LOCATIONS FOR SOURCE PROGRAM FORMAT STATEMENTS

	EFN	LOC		EFN	LOC		EFN	LOC		EFN	LOC
8)2T	93	04256	8)2U	94	04253	8)2V	95	04241	8)30	96	04210
8)31	97	04200	8)32	98	04167	8)33	99	04157	8)43	131	04145
8)44	132	04133	8)46	134	04121						

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	OEC	OCT		OEC	OCT		OEC	OCT		DEC	OCT
1)	2226	04262	2)	2094	04056	3)	2099	04063	4)	32767	77777
6)	2109	04075	9)	2223	04257	C)G1	2236	04274	C)G2	2237	04275
C)G3	2238	04276	C)G4	2239	04277	C)G5	2240	04300	C)G6	2241	04301
C)G7	2242	04302	C)G8	2243	04303	C)G9	2244	04304	C)GA	2245	04305
C)GB	2246	04306	C)GC	2247	04307	C)GO	2248	04310	C)GE	2249	04311
C)GF	2250	04312	C)GG	2251	04313	C)GH	2252	04314	C)GI	2253	04315
C)GJ	2254	04316	C)203	2255	04317	C)204	2256	04320	C)205	2257	04321

C)206	2258	04322	C)207	2259	04323	C)208	2260	04324	C)20A	2261	04325
C)208	2262	04326	C)20C	2263	04327	C)20E	2264	04330	C)20G	2265	04331
C)20H	2266	04332	C)20I	2267	04333	C)20K	2268	04334	C)20M	2269	04335
C)20O	2270	04336	C)20Q	2271	04337	D)10S	678	01246	D)10U	765	01375
D)11A	842	01512	D)11D	858	01532	D)11F	867	01543	D)11Q	960	01700
D)136	1630	03136	D)13P	1840	03460	D)20R	673	01241	D)211	795	01433
D)21M	918	01626	D)22A	1247	02337	D)231	1583	03057	D)23V	1895	03547
D)241	1940	03624	D)243	1956	03644	D)24G	2089	04051	D)31A	841	01511
D)32A	1246	02336	D)331	1582	03056	D)336	1629	03135	D)33V	1894	03546
D)341	1939	03623	D)343	1955	03643	D)34G	2088	04050	D)40M	576	01100
D)40N	636	01174	D)40T	748	01354	D)416	825	01471	D)41C	850	01522
D)426	1139	02163	D)427	1229	02315	D)42F	1295	02417	D)42P	1431	02627
D)42V	1539	03003	D)43D	1708	03254	D)43H	1744	03320	D)43L	1775	03357
D)43M	1796	03404	D)43N	1803	03413	D)51D	857	01531	D)526	1138	02162
D)527	1228	02314	D)52F	1294	02416	D)52V	1538	03002	D)53N	1802	03412
D)53P	1839	03457	D)60N	635	01173	D)61C	849	01521	D)62P	1430	02626
D)63D	1707	03253	D)63L	1774	03356	D)63M	1795	03403	D)71A	840	01510
D)71D	856	01530	D)73P	1838	03456	E)F	533	01025	E)I	543	01037
E)J	548	01044	E)11	792	01430	E)13	803	01443	E)16	823	01467
E)18	834	01502	E)1I	888	01570	E)1L	910	01616	E)1O	948	01664
E)2T	1529	02771	E)3A	1677	03215	E)3M	1794	03402	E)45	1966	03656
E)48	1986	03702	E)48	2039	03767	E)41Q	957	01675			

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
CLOSE	7	00007	OPMLTX	5	00005	MPRINT	11	00013	NPNRMX	6	00006
SWEEPX	8	00010	(FIL)	10	00012	(RLR)	2	00002	(RWT)	0	00000
(STB)	3	00003	(STH)	9	00011	(TSB)	1	00001	(WLR)	4	00004

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

CLOSE	OPMLTX	MPRINT	NPNRMX	SWEEPX	(FIL)	(RLR)	(RWT)
(STB)	(STH)	(TSB)	(WLR)				

EXTERNAL FORMULA NUMBERS WITH CORRESPONOING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
3	22	00672	4	23	00674	5	26	00710	1	29	00723
6	36	00751	2	38	00762	7	43	01005	8	44	01011
9	48	01026	10	49	01030	11	50	01034	12	51	01040
13	52	01045	14	56	01072	15	57	01101	16	73	01226
17	74	01242	18	75	01247	19	86	01401	20	91	01441
21	93	01451	22	96	01500	23	97	01503	100	98	01513
102	100	01516	104	101	01523	24	103	01533	25	104	01537
31	105	01544	26	106	01546	261	110	01573	27	111	01600
28	112	01613	29	118	01651	30	119	01665	32	122	01701
33	128	01742	36	131	01761	361	138	02035	37	139	02054
38	140	02075	35	148	02164	39	151	02316	40	154	02330
41	159	02372	44	161	02410	45	165	02444	46	167	02455
113	172	02504	49	173	02506	114	176	02604	50	178	02611
52	181	02620	54	192	02725	123	197	02772	116	199	02777
117	200	03004	53	206	03034	531	212	03104	55	214	03115
56	218	03137	58	223	03201	106	228	03243	124	229	03246
107	230	03255	118	231	03261	119	233	03264	120	237	03312
125	241	03343	59	243	03360	62	246	03375	108	247	03405
109	248	03414	121	250	03420	70	257	03461	71	258	03463
73	259	03470	72	273	03611	75	274	03625	80	275	03631
81	279	03653	82	281	03657	87	283	03666	85	285	03700
83	286	03703	84	290	03730	86	294	03764	88	296	03774
90	301	04021	92	309	04052						

```
C     CALL NPNRMX,(A, B, N, FL, INDEX, MD, NX, NP )                      MTRS0135
C            A=VECTOR TO BE NORMALIZED          B=NORMALIZED VECTOR(MAY=A)MTRS0136
C            N=SIZE                             FL=NORMALIZING NUMBER     MTRS0137
C            INDEX=+ ON ENTRY, NORMALIZE ON NUMBER WHOSE INDEX IS INDEX MTRS0138
C                 =0 ON ENTRY, NORMALIZE ON LARGEST         S.H. AND SET MTRS0139
C                                     INDEX=TO ITS INDEX.                MTRS0140
C                 =- ON ENTRY, NORMALIZE ON FL.                          MTRS0141
C            MD=SINGLE PRECISION DIMENSIONED NUMBER OF ROWS OF A ANO B  MTRS0142
C            NX=1, VECTOR REAL                                           MTRS0143
C              =2, VECTOR COMPLEX                                        MTRS0144
C            NP=1, SINGLE PRECISION                                      MTRS0145
C              =2, DOUBLE PRECISION                                      MTRS0146
                                                                         MTRS0147
      SUBROUTINE  NPNRMX (A, B, N, FL, INDEX, MO, NX, NP )              MTRS0148
                                                                         MTRS0149
D     DIMENSION  A(1), B(1), FL(1), D(1), C(1)                          MTRS0150
                                                                         MTRS0151
      N1=1+NP                                                            MTRS0152
      N2=N*NP                                                            MTRS0153
      N4=MD*NP                                                           MTRS0154
      IF  ( INDEX )   32, 7,38                                           MTRS0155
    7 GOTO  (11,8),NX                                                    MTRS0156
    8 FL= (A(1)**2+A(N4+1)**2)                                           MTRS0157
      INDEX=1                                                            MTRS0158
      DO  10  K=N1,N2,NP                                                 MTRS0159
        I=K+N4                                                           MTRS0160
        D= (A(K)**2+A(I)**2)                                             MTRS0161
        IF  ( FL-D )   9,9,10                                            MTRS0162
    9   FL=0                                                             MTRS0163
        INOEX=K                                                          MTRS0164
   10   CONTINUE                                                         MTRS0165
    6 FL=A(INDEX)                                                        MTRS0166
      GOTO  (18,25),NP                                                   MTRS0167
                                                                         MTRS0168
   11 FL=ABSF(A(1))                                                      MTRS0169
      INDEX=1                                                            MTRS0170
      DO  13  K=N1,N2,NP                                                 MTRS0171
```

```
      D=ABSF(A(K))                                              MTRS0172
      IF  (FL-D)     12,12,13                                   MTRS0173
 12   FL=D                                                      MTRS0174
      INDEX=K                                                   MTRS0175
 13   CONTINUE                                                  MTRS0176
                                                                MTRS0177
 14 FL=A(INDEX)                                                 MTRS0178
    GOTO (16,21),NP                                             MTRS0179
 16 DO  17  I=1,N                                               MTRS0180
 17   B(I)=A(I)/FL                                              MTRS0181
    GOTO  30                                                    MTRS0182
                                                                MTRS0183
 18 I=INDEX+MD                                                  MTRS0184
    FL(2)=A(I)                                                  MTRS0185
 19 D=FL(1)**2+FL(2)**2                                         MTRS0186
    DO  20  I=1,N                                               MTRS0187
      K=I+MD                                                    MTRS0188
      C=A(I)*FL(2)-A(K)*FL(1)                                   MTRS0189
      B(I)=(A(I)*FL(1)+A(K)*FL(2))/D                            MTRS0190
 20   B(K)=-C/D                                                 MTRS0191
    GOTO  30                                                    MTRS0192
                                                                MTRS0193
 21 FL(2)=A(INDEX+1)                                            MTRS0194
 23 DO  24  I=1,N2,NP                                           MTRS0195
D 24  B(I)=A(I)/FL                                              MTRS0196
    GOTO  26                                                    MTRS0197
                                                                MTRS0198
 25 FL(2)=A(INDEX+1)                                            MTRS0199
    I=INDEX+N4                                                  MTRS0200
D   FL(3)=A(I)                                                  MTRS0201
D 26 D=FL(1)**2+FL(3)**2                                        MTRS0202
    DO  27  I=1,N2,NP                                           MTRS0203
      K=I+N4                                                    MTRS0204
D     C=A(I)*FL(3)-A(K)*FL(1)                                   MTRS0205
D     B(I)=(A(I)*FL(1)+A(K)*FL(3))/D                            MTRS0206
D 27  B(K)=-C/D                                                 MTRS0207
                                                                MTRS0208
 28 INDEX=INDEX/2+1                                             MTRS0209
```

```
                                                          MTRS0210
30 RETURN                                                 MTRS0211
                                                          MTRS0212
32 GOTO  (34,36), NX                                      MTRS0213
34 GOTO (16,23),NP                                        MTRS0214
36 GOTO  (19,26),NP                                       MTRS0215
                                                          MTRS0216
38 GOTO  (40,39),NP                                       MTRS0217
39 INDEX=2*INDEX-1                                        MTRS0218
40   GOTO  (14,6),NX                                      MTRS0219
   END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

```
                                   STORAGE NOT USED BY PROGRAM

      DEC   OCT           DEC   OCT
      590 01116         32561 77461

                    STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

      DEC   OCT              DEC   OCT              DEC   OCT              DEC   OCT
  C   587 01113         D    589 01115

 STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

      DEC   OCT              DEC   OCT              DEC   OCT              DEC   OCT
  I   585 01111         K    584 01110        N1    583 01107        N2    582 01106
 N4   581 01105

                         LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

      DEC   OCT              DEC   OCT              DEC   OCT              DEC   OCT
  1)  571 01073        2)    560 01060        4)  32767 77777        6)    562 01062
  9)  568 01070      C)G0    575 01077      C)G2    576 01100      C)G3    577 01101
C)G4  578 01102      C)G5    579 01103      C)G6    580 01104      D)103   220 00334
D)107 251 00373      D)10D   287 00437      D)10E   295 00447      D)10I   312 00470
D)10M 357 00545      D)10U   531 01023      D)214   556 01054      D)407   249 00371
D)607 248 00370       E)2    200 00310       E)5    243 00363       E)6    247 00367
 E)V  536 01030       E)12   545 01041      E)10E   292 00444

                                LOCATIONS OF NAMES IN TRANSFER VECTOR

       DEC   OCT             DEC   OCT              DEC   OCT              DEC   OCT
DEXP(2   1 00001     (DFAD)    2 00002     (DFDP)     0 00000     (DFMP)     3 00003
(DFSB)   4 00004

                         ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY
```

DEXP(2 (DFAO) (OFDP) (DFMP) (DFSB)

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
7	8	00305	8	9	00311	9	15	00354	10	17	00364
6	18	00374	11	20	00401	12	25	00425	13	27	00433
14	28	00440	16	30	00430	17	31	00453	18	33	00461
19	35	00471	20	40	00531	21	42	00542	23	43	00546
24	44	00553	25	46	00574	26	49	00612	27	54	00760
28	55	01011	30	56	01024	32	58	01031	34	59	01033
36	60	01036	38	61	01042	39	62	01045	40	63	01055

```
C                                                                          MTRS0222
C COMPUTES TRUE MODE AND SWEEPS IT FROM THE MATRIX. (REAL OR COMPLEX)     MTRS0223
C                                                                          MTRS0224
C  HTRUE = TRUE MODAL COLUMNS, AS COMPUTED. U = DYNAMIC MATRIX.            MTRS0225
C      H = SERIES OF MODIFIED MODAL COLUMNS.FL= COLUMN OF EIGENVALUES.     MTRS0226
C     US = SERIES OF MODIFIED MODAL ROWS OF U.                             MTRS0227
C   MODE = MODE NOW BEING COMPUTED.          N = SIZE                      MTRS0228
C     MD = DIMENSIONED NUMBER OF ROWS OF U,US,H,HTRUE                      MTRS0229
C     NX = 1 IF PROBLEM IS REAL.                                           MTRS0230
C        = 2 IF PROBLEM IS COMPLEX.                                        MTRS0231
                                                                           MTRS0232
      SUBROUTINE SWEEPX (HTRUE, U,H, US,FL, MODE, N, MD, NC, INDEX, EP)    MTRS0233
                                                                           MTRS0234
      DIMENSION  H(1), US(1), U(1), HTRUE(1), FL(1), G(4)                  MTRS0235
                                                                           MTRS0236
      M=MODE-1                                                             MTRS0237
      K1=M*NC*MD                                                           MTRS0238
                                                                           MTRS0239
      DO   6  J=1,NC                                                       MTRS0240
        K=K1+(J-1)*MD                                                      MTRS0241
        DO   6  L=1,N                                                      MTRS0242
          K=K+1                                                            MTRS0243
    6     HTRUE(K)=H(K)                                                    MTRS0244
                                                                           MTRS0245
      IF  ( M )  31,31,8                                                   MTRS0246
    8 DO  25  I=1,M                                                        MTRS0247
        L1=NC*MD*(MODE-I) -NC*MD                                           MTRS0248
      GOTO  ( 9,11),NC                                                     MTRS0249
    9 G=0.                                                                 MTRS0250
      DO  10  J=1,N                                                        MTRS0251
        L=L1+J                                                             MTRS0252
        K=K1+J                                                             MTRS0253
   10   G=G+US(L)*HTRUE(K)                                                 MTRS0254
      GOTO  13                                                             MTRS0255
   11           G(1)=0.                                                    MTRS0256
                G(2)=0.                                                    MTRS0257
      DO  12  J1=1,N                                                       MTRS0258
```

```
      L=L1+J1                                                           MTRS0259
      K=K1+J1                                                           MTRS0260
      L2=L+MD                                                           MTRS0261
      K2=K+MD                                                           MTRS0262
      G(1)=G(1)+US(L)*HTRUE(K)-US(L2)*HTRUE(K2)                         MTRS0263
12    G(2)=G(2)+US(L)*HTRUE(K2)+US(L2)*HTRUE(K)                         MTRS0264
13 K=MODE-I                                                             MTRS0265
   GOTO (14,19),NC                                                      MTRS0266
14 IF (ABSF(FL(K)/FL(MODE)-1.) - EP )          15,15,16                 MTRS0267
15 G=1.                                                                 MTRS0268
   GOTO  17                                                             MTRS0269
16 G=(FL(K)-FL(MODE)) / G                                               MTRS0270
17 DO  18  J=1,N                                                        MTRS0271
       K=K1+J                                                           MTRS0272
       L=L1+J                                                           MTRS0273
18     HTRUE(K)=H(L)-G(1)*HTRUE(K)                                      MTRS0274
     GOTO  25                                                           MTRS0275
19 K=2*K                                                                MTRS0276
   J=2*MODE                                                             MTRS0277
   IF ( ABSF((FL(K-1)*FL(J-1)+FL(K)*FL(J))/(FL(J-1)**2+FL(J)**2)-1.)    MTRS0278
  1            -EP)        20,20,22                                     MTRS0279
20 IF ( ABSF((FL(K)*FL(J-1)-FL(K-1)*FL(J)) / (FL(J-1)**2+FL(J)**2))     MTRS0280
  1            -EP)        21,21,22                                     MTRS0281
21 G(1)=1.                                                              MTRS0282
   G(2)=0.                                                              MTRS0283
   GOTO  23                                                             MTRS0284
22 G(3)=G(1)**2+G(2)**2                                                 MTRS0285
     G(4)=(FL(K)-FL(J))*G(1)-(FL(K-1)-FL(J-1))*G(2)                     MTRS0286
     G(1)=((FL(K-1)-FL(J-1))*G(1)+(FL(K)-FL(J))*G(2)) / G(3)            MTRS0287
     G(2)= G(4) / G(3)                                                  MTRS0288
                                                                        MTRS0289
23   DO  24  J1=1,N                                                     MTRS0290
       K=K1+J1                                                          MTRS0291
       K2=K+     MD                                                     MTRS0292
       L=L1+J1                                                          MTRS0293
       L2=L+MD                                                          MTRS0294
       G(3)=HTRUE(K)                                                    MTRS0295
       HTRUE(K)= H(L)+    G(2)*HTRUE(K2)-G(1)*HTRUE(K)                  MTRS0296
```

```
      HTRUE(K2)= H(L2)- G(1)*HTRUE(K2)-G(2)*G(3)                MTRS0297
24    CONTINUE                                                  MTRS0298
25  CONTINUE                                                    MTRS0299
                                                                MTRS0300
    I=0                                                         MTRS0301
    CALL NPNRMX  (HTRUE(K1+1),HTRUE(K1+1),N,G,I,MD,NC,1)        MTRS0302
                                                                MTRS0303
31 GOTO  (26,32),NC                                             MTRS0304
26 DO  29  J=1,N                                                MTRS0305
     L1=(J-1)*MD                                                MTRS0306
     L2=K1+J                                                    MTRS0307
   DO  29  I=1,N                                                MTRS0308
       L=L1+I                                                   MTRS0309
     IF  (I-INDEX)  28,27,28                                    MTRS0310
27   U(L)=0.                                                    MTRS0311
     GOTO  29                                                   MTRS0312
28     K=K1+I                                                   MTRS0313
       U(L)=U(L)-H(K)*US(L2)                                    MTRS0314
29     CONTINUE                                                 MTRS0315
                                                                MTRS0316
30 RETURN                                                       MTRS0317
                                                                MTRS0318
32 DO  35  I=1,N                                                MTRS0319
     L1=MD*NC*(I-1)                                             MTRS0320
     L2=K1+I                                                    MTRS0321
       J=L2+MD                                                  MTRS0322
     DO  35  J1=1,N                                             MTRS0323
       L=L1+J1                                                  MTRS0324
       K3=L+MO                                                  MTRS0325
       IF  (J1-INDEX)    34,33,34                               MTRS0326
33     U(L)=0.                                                  MTRS0327
       U(K3)=0.                                                 MTRS0328
   GOTO 35                                                      MTRS0329
34     K=K1+J1                                                  MTRS0330
       K2=K+MD                                                  MTRS0331
       U(L)=U(L)-H(K)*US(L2)+H(K2)*US(J)                        MTRS0332
       U(K3)=U(K3)-H(K2)*US(L2)-H(K)*US(J)                      MTRS0333
35     CONTINUE                                                 MTRS0334
```

```
      GOTO 30                                                           MTRS0335
                                                                        MTRS0336
      END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)                                MTRS0337
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
744	01350	32561	77461

STORAGE LOCATIONS FOR VARIABLES APPEARING IN DIMENSION AND EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
G	743	01347									

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
I	739	01343	J1	738	01342	J	737	01341	K1	736	01340
K2	735	01337	K3	734	01336	K	733	01335	L1	732	01334
L2	731	01333	L	730	01332	M	729	01331			

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
1)	706	01302	2)	692	01264	3)	695	01267	6)	697	01271
9)	703	01277	C)G0	710	01306	C)G1	711	01307	C)G2	712	01310
C)G4	713	01311	C)G5	714	01312	C)G6	715	01313	C)G7	716	01314
C)G8	717	01315	C)G9	718	01316	C)201	719	01317	C)202	720	01320
C)203	721	01321	C)204	722	01322	C)205	723	01323	C)206	724	01324
C)207	725	01325	C)208	726	01326	C)209	727	01327	C)20A	728	01330
D)110	551	01047	D)112	565	01065	D)206	207	00317	D)200	305	00461
D)20T	525	01015	D)21C	680	01250	D)40R	496	00760	D)413	577	01101
D)416	539	01115									

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
NPNRMX	0	00000									

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

NPNRMX

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
6	10	00276	8	12	00311	9	15	00341	10	19	00363
11	21	00373	12	29	00442	13	30	00462	14	32	00472
15	33	00503	16	35	00506	17	36	00512	18	39	00532
19	41	00543	20	44	00603	21	45	00626	22	48	00633
23	52	00674	24	60	00753	25	61	00761	31	65	01016
26	66	01020	27	72	01062	28	74	01066	29	76	01102
30	77	01116	32	79	01122	33	87	01200	34	90	01206
35	94	01251									

```
                                                                        MTRS0707
C                                                                       
C   SUBROUTINE FINDS 2 CLOSE EIGENVALUESAND THEIR ASSOCIATED EIGENVECTORSMTRS0708
C        OF A MATRIX ON WHICH ITERATIONIS IN PROGRESS.                  MTRS0709
C                                                                       MTRS0710
C    U = DYNAMIC MATRIX                    EIG1=1ST. EIGENVALUE         MTRS0711
C    GUESS=1ST. GUESS VECTOR               EIG2=2ND. EIGENVALUE         MTRS0712
C        (RESULT OF K PRIOR ITERATIONS)       N=SIZE                    MTRS0713
C    H1=1ST. VECTOR SOLUTION               MAXITR=MAXIMUM NO. ITERATIONS MTRS0714
C    H2=2ND. VECTOR SOLUTION               ITERS=ACTUAL NO. OF ITERATIONS MTRS0715
C    INDEX = NORMALIZING NUMBER, 1ST. EIGENVALUE                        MTRS0716
C    INDEX1= NORMALIZING NUMBER, 2ND. EIGENVALUE                        MTRS0717
C    MAXR=SINGLE PRECISION DIMENSIONED NO. OF ROWS OF U,H1,H2,GUESS.    MTRS0718
C    NC = 1, MATRIX REAL                                                MTRS0719
C       = 2, MATRIX COMPLEX                                             MTRS0720
C    H= BLOCK OF WORKING CORE(2*MAXR-REAL, 4*MAXR-COMPLEX)*2            MTRS0721
C    FL=20 CORE STORAGES FOR TEMPORARY RESULTS                          MTRS0722
C    TEST = TEST NUMBER FOR DETERMINING CONVERGENCE OF P AND Q          MTRS0723
C    AITKEN = TEST NUMBER FOR DETERMINING CONVERGENCE OF SPEEDER-UPPER  MTRS0724
                                                                        MTRS0725
      SUBROUTINE CLOSE (U, GUESS, H1, H2, EIG1, EIG2, INDEX, INDEX1, N, MTRS0726
     1                  NC, MAXITR, ITERS, NAKDR, TEST, TESTSR, AITKEN, MTRS0727
     2                  AT, MAXR, H, FL, IR)                            MTRS0728
                                                                        MTRS0729
D     DIMENSION  U(1),GUESS(1),H1(1),H2(1),H(1),FL(1),TEST(1),AITKEN(1) MTRS0730
     1           ,EIG1(1), EIG2(1)                                      MTRS0731
D     DIMENSION  P(1), PI(1), Q(1), QI(1), P1(1), P1I(1), Q1(1), Q1I(1) MTRS0732
                                                                        MTRS0733
      IF ACCUMULATOR OVERFLOW       5,5                                 MTRS0734
    5 IF DIVIDE CHECK           6,6                                     MTRS0735
    6 IR=0                                                              MTRS0736
      NAK=4                                                             MTRS0737
      ITERS=0                                                           MTRS0738
      MX=NC*N                                                           MTRS0739
      NX=2*N                                                            MTRS0740
      CALL OPFORM  (0,U,U,0,N,MX,MAXR,MAXR)                             MTRS0741
      CALL OPFORM  (0,GUESS,GUESS,0,N,NC,MAXR,MAXR)                     MTRS0742
      IGOTO=1                                                           MTRS0743
```

```
      K1=2*NC*MAXR                                                      MTRS0744
                                                                        MTRS0745
    8 ITERS = ITERS+1                                                   MTRS0746
      INDEX=0                                                           MTRS0747
      I1=4*(NAK-1)+1                                                    MTRS0748
      CALL OPMLTX  (U,NC,GUESS,NC,H,N,N,1,MAXR,MAXR,MAXR,2)             MTRS0749
      CALL NPNRMX (H,GUESS,N,FL(I1),INDEX,MAXR,NC,2)                    MTRS0750
                                                                        MTRS0751
      IF  ACCUMULATOR OVERFLOW      84,9                                MTRS0752
    9 IF OIVIOE CHECK       84,10                                       MTRS0753
                                                                        MTRS0754
   10 GOTO  (11,45,49,51),NAK                                           MTRS0755
   11 GOTO  (12,35),NC                                                  MTRS0756
   12 I1=2*INOEX-1                                                      MTRS0757
      I2=I1+K1                                                          MTRS0758
D     P1=FL(5)*(  (FL(9)*H2(I1)*H1(I1)-FL(1)*GUESS(I1)*H(I2)) / (FL(5)* MTRS0759
     1            H2(I1)*H(I2)-FL(9)*H1(I1)**2)  )                      MTRS0760
0     Q1=FL(5)*FL(9)*(  (FL(1)*GUESS(I1)*H1(I1)-FL(5)*H2(I1)**2) /      MTRS0761
     1            (FL(5)*H2(I1)*H (I2)-FL(9)*H1(I1)**2) )               MTRS0762
0     P1I=0.                                                            MTRS0763
D     Q1I=0.                                                            MTRS0764
      FL(19)=(FL(1)-FL(5))**2                                           MTRS0765
      GDTO  (56,13),IGOTO                                               MTRS0766
                                                                        MTRS0767
   13 FL(17) = ( SQRTF( ((P1-P)**2+(P1I-PI)**2)/( SQRTF(Q1**2+Q1I**2))))MTRS0768
      FL(18) = ABSF (1.-SQRTF( (Q1**2+Q1I**2) / (Q**2+QI**2) ) )        MTRS0769
                                                                        MTRS0770
      IF OIVIOE CHECK        22,22                                      MTRS0771
   22 IF  ( FL(17)-TEST )   14,14,15                                    MTRS0772
   14 IF  ( FL(18)-TEST )   60,60,15                                    MTRS0773
   15 OO  16  J=1,NC                                                    MTRS0774
        J1=2*MAXR*(J-1)                                                 MTRS0775
        OO  16  I=1,NX,2                                                MTRS0776
          K=J1+I                                                        MTRS0777
          IF  ( ABSF(H2(K)-GUESS(K)) - TESTSR)    16,16,17              MTRS0778
   16     CONTINUE                                                      MTRS0779
      GOTO  82                                                          MTRS0780
                                                                        MTRS0781
```

```
   17 GOTO  (56,18),IGOTO                                                 MTRS0782
   18 GOTO  (19,40),NC                                                    MTRS0783
                                                                          MTRS0784
   19 IF  (MAXITR-ITERS-5)  54,54,190                                     MTRS0785
  190 DO  21  I=1,NX,2                                                    MTRS0786
        IF  (H2(I)-H1(I))   20,191,20                                     MTRS0787
  191   IF  ( (GUESS(I)-H2(I)) )        54,21,54                          MTRS0788
   20   IF  (ABSF( (GUESS(I)-H2(I))/(H2(I)-H1(I)) ) - AITKEN) 21,21,54    MTRS0789
   21   CONTINUE                                                          MTRS0790
                                                                          MTRS0791
      DO  28  I=1,NX,2                                                    MTRS0792
D       Q= (GUESS(I)-2.*H2(I)+H1(I))                                      MTRS0793
        IF ( Q )   25,24,25                                               MTRS0794
D  24   H(I)=H1(I)                                                        MTRS0795
        GOTO  28                                                          MTRS0796
D  25   H(I)= H1(I)-( (H2(I)-H1(I))**2) / Q                               MTRS0797
   28 CONTINUE                                                            MTRS0798
   29 NAK=5                                                               MTRS0799
      NAKDR=NAKDR+1                                                       MTRS0800
      IGOTO=1                                                             MTRS0801
                                                                          MTRS0802
C EXCHANGE TRIAL VECTOR AND TEST FOR MAXIMUM NUMBER OF ITERATIONS.        MTRS0803
   30 DO  31  J=1,NC                                                      MTRS0804
        J1=MAXR*(J-1)                                                     MTRS0805
        DO  31  I=1,NX                                                    MTRS0806
          K=J1+I                                                          MTRS0807
   31     GUESS(K)=H(K)                                                   MTRS0808
   32 NAK=NAK-1                                                           MTRS0809
   33 IF ( ITERS-MAXITR )   8,85,85                                       MTRS0810
                                                                          MTRS0811
C COMPLEX P1 AND Q1 COMPUTATIONS                                          MTRS0812
   35 I1=2*INDEX-1                                                        MTRS0813
      I2=I1+K1                                                            MTRS0814
      I3=I1+2*MAXR                                                        MTRS0815
      I4=I2+2*MAXR                                                        MTRS0816
D     X=FL(1)*GUESS(I1)-FL(3)*GUESS(I3)                                   MTRS0817
D     X1=FL(9)*H1(I1)-FL(11)*H1(I3)                                       MTRS0818
D     X2=FL(3)*GUESS(I1)+FL(1)*GUESS(I3)                                  MTRS0819
```

```
D      X3=FL(11)*H1(I1)+FL(9)*H1(I3)                                         MTRS0820
D      X4=FL(5)*H2(I1)-FL(7)*H2(I3)                                          MTRS0821
D      X5=FL(7)*H2(I1)+FL(5)*H2(I3)                                          MTRS0822
D      X6=H(I2)*X4-H(I4)*X5-H1(I1)*X1+H1(I3)*X3                              MTRS0823
D      X7=H(I2)*X5+H(I4)*X4-H1(I1)*X3-H1(I3)*X1                              MTRS0824
                                                                             MTRS0825
D      X8= FL(5)*(H2(I1)*X1-H2(I3)*X3-H(I2)*X+H(I4)*X2) - FL(7)*(H2(I3)     MTRS0826
     1          *X1+H2(I1)*X3-H(I4)*X-H(I2)*X2)                              MTRS0827
D      X9= FL(5)*(H2(I3)*X1+H2(I1)*X3-H(I4)*X-H(I2)*X2) + FL(7)*(H2(I1)     MTRS0828
     1          *X1-H2(I3)*X3-H(I2)*X+H(I4)*X2)                              MTRS0829
D      P1=  (X8*X6+X9*X7) / (X6**2+X7**2)                                    MTRS0830
D      P1I= (X9*X6-X8*X7) / (X6**2+X7**2)                                    MTRS0831
                                                                             MTRS0832
D      X8=  (FL(5)*FL(9)-FL(7)*FL(11))*(H1(I1)*X-H1(I3)*X2-H2(I1)*X4+       MTRS0833
     1           H2(I3)*X5) - (FL(7)*FL(9)+FL(5)*FL(11))*(H1(I3)*X+          MTRS0834
     2           H1(I1)*X2-H2(I3)*X4-H2(I1)*X5)                              MTRS0835
D      X9=(FL(5)*FL(9)-FL(7)*FL(11))*(H1(I3)*X+H1(I1)*X2-H2(I3)*X4-         MTRS0836
     1           H2(I1)*X5) + (FL(7)*FL(9)+FL(5)*FL(11))*(H1(I1)*X-          MTRS0837
     2           H1(I3)*X2-H2(I1)*X4+H2(I3)*X5)                              MTRS0838
D      Q1= (X8*X6+X9*X7) / (X6**2+X7**2)                                     MTRS0839
D      Q1I=(X9*X6-X8*X7) / (X6**2+X7**2)                                     MTRS0840
       FL(19)=(FL(1)-FL(5))**2+(FL(3)-FL(7))**2                              MTRS0841
       GOTO  (56,13),IGOTO                                                   MTRS0842
  40 IF  (MAXITR-ITERS-5)  54,54,400                                         MTRS0843
 400 DO  41  I=1,NX,2                                                        MTRS0844
         J1=2*MAXR+I                                                         MTRS0845
         X=(H2(I)-H1(I))**2 + (H2(J1)-H1(J1))**2                             MTRS0846
         IF ( X )              37,401,37                                     MTRS0847
 401     IF ( (GUESS(I)-H2(I))**2 + (GUESS(J1)-H2(J1))**2 )   54,41,54      MTRS0848
  37     IF ( ( (GUESS(I)-H2(I))**2 + (GUESS(J1)-H2(J1))**2 ) /X -AT )      MTRS0849
     1                         41,41,54                                      MTRS0850
  41     CONTINUE                                                            MTRS0851
       DO  44  I=1,NX,2                                                      MTRS0852
         J=2*MAXR+I                                                          MTRS0853
D        X=GUESS(I)-2.*H2(I)+H1(I)                                           MTRS0854
         IF ( X )  43,42,43                                                  MTRS0855
D 42     H(I)=H1(I)                                                          MTRS0856
D        H(J)=H1(J)                                                          MTRS0857
```

```
         GOTO  44                                                      MTRS0858
D  43    X1=GUESS(J)-2.*H2(J)+H1(J)                                    MTRS0859
D        X2=H2(I)-H1(I)                                                MTRS0860
D        X3=H2(J)-H1(J)                                                MTRS0861
D        H(I) = H1(I) - ((X2**2-X3**2)*X+(2.*X2*X3*X1))/(X**2+X1**2)   MTRS0862
D        H(J) = H1(J) - ((2.*X2*X3*X)-(X2**2-X3**2)*X1) / (X**2+X1**2) MTRS0863
   44 CONTINUE                                                         MTRS0864
      GOTO  29                                                         MTRS0865
                                                                       MTRS0866
   45 DO  46  J=1,NC                                                   MTRS0867
        J1=MAXR*(J-1)*2                                                MTRS0868
        DO  46  I=1,NX                                                 MTRS0869
        K=J1+I                                                         MTRS0870
   46   H2(K)=GUESS(K)                                                 MTRS0871
      GOTO  32                                                         MTRS0872
   49  DO  50  J=1,NC                                                  MTRS0873
        J1=MAXR*(J-1) *2                                               MTRS0874
        DO  50  I=1,NX                                                 MTRS0875
          K=J1+I                                                       MTRS0876
   50     H1(K)=GUESS(K)                                               MTRS0877
      GOTO  32                                                         MTRS0878
                                                                       MTRS0879
   51 DO  52  J=1,NC                                                   MTRS0880
        J1=2*MAXR*(J-1)                                                MTRS0881
        DO  52  I=1,NX                                                 MTRS0882
          K=J1+I                                                       MTRS0883
          L=K1+I +J1                                                   MTRS0884
   52     H(L)=GUESS(K)                                                MTRS0885
      GOTO  32                                                         MTRS0886
                                                                       MTRS0887
   54 IF  ( ITERS-MAXITR )   56,85,85                                  MTRS0888
                                                                       MTRS0889
   56 DO  58  J=1,NC                                                   MTRS0890
        J1=MAXR*(J-1)*2                                                MTRS0891
        DO  58  I=1,NX                                                 MTRS0892
          K=J1+I                                                       MTRS0893
       L=K1+J1+I                                                       MTRS0894
       H(L)=H1(K)                                                      MTRS0895
```

```
         H1(K)=H2(K)                                           MTRS0896
   58    H2(K)=GUESS(K)                                        MTRS0897
      IGOTO=2                                                  MTRS0898
      J=17                                                     MTRS0899
      DO  59  I=1,12                                           MTRS0900
        J=J-1                                                  MTRS0901
   59   FL(J)=FL(J-4)                                          MTRS0902
D     P=P1                                                     MTRS0903
D     Q=Q1                                                     MTRS0904
D     PI =P1I                                                  MTRS0905
D     QI =Q1I                                                  MTRS0906
      GOTO  33                                                 MTRS0907
   60 GOTO  (64,70),NC                                         MTRS0908
                                                               MTRS0909
D  64 P= SQRTF( P1**2-4.*Q1)                                   MTRS0910
D     FL(9) =(-P1+P) / 2.                                      MTRS0911
D     FL(5) =(-P1-P) / 2.                                      MTRS0912
D     IF  ( ABSF(FL(9)) - ABSF(FL(5)) )          65,66,66      MTRS0913
D  65 P=FL(5)                                                  MTRS0914
D     FL(5)=FL(9)                                              MTRS0915
D     FL(9)=P                                                  MTRS0916
   66 DO  68  I=1,NX,2                                         MTRS0917
D       H1(I)=FL(5)*H2(I)-FL(1)*GUESS(I)                       MTRS0918
D  68   H2(I)=FL(1)*GUESS(I)-FL(9)*H2(I)                       MTRS0919
      GOTO  76                                                 MTRS0920
                                                               MTRS0921
D  70 X=(P1**2-4.*Q1-P1I**2)                                   MTRS0922
D     X1=(2.*P1*P1I-4.*Q1I)                                    MTRS0923
      IF  ( X )              73,75,75                          MTRS0924
D  73 PI= SQRTF( (-X + SQRTF( X**2 + X1**2 )) / 2.)            MTRS0925
D     P = X1 / (2.*PI)                                         MTRS0926
      GOTO  77                                                 MTRS0927
D  75 P = SQRTF( ( X + SQRTF( X**2 + X1**2 )) / 2.)            MTRS0928
D     PI= X1 / (2.*P)                                          MTRS0929
D  77 FL(9) = (-P1+P) / 2.                                     MTRS0930
D     FL(11)= (-P1I+PI) / 2.                                   MTRS0931
D     FL(5) = (-P1-P) / 2.                                     MTRS0932
D     FL(7) = (-P1I-PI) / 2.                                   MTRS0933
```

```
D      IF  ( (FL(9)**2+FL(11)**2) - (FL(5)**2+FL(7)**2) )      71,72,72    MTRS0934
D   71 P=FL(5)                                                             MTRS0935
D      FL(5)=FL(9)                                                         MTRS0936
D      FL(9)=P                                                             MTRS0937
D      P=FL(7)                                                             MTRS0938
D      FL(7)=FL(11)                                                        MTRS0939
D      FL(11)=P                                                            MTRS0940
    72 DD  74  I=1,NX,2                                                    MTRS0941
       J=2*MAXR+I                                                          MTRS0942
D      H1(I) = FL(5)*H2(I)-FL(7)*H2(J)-FL(1)*GUESS(I)+FL(3)*GUESS(J)       MTRS0943
D      H1(J) = FL(5)*H2(J)+FL(7)*H2(I)-FL(1)*GUESS(J)-FL(3)*GUESS(I)       MTRS0944
D      P=FL(11)*H2(I)                                                      MTRS0945
D      H2(I) = FL(1)*GUESS(I)-FL(3)*GUESS(J)-FL(9)*H2(I)+FL(11)*H2(J)      MTRS0946
D   74 H2(J) = FL(3)*GUESS(I)+FL(1)*GUESS(J)-FL(9)*H2(J)-P                 MTRS0947
                                                                           MTRS0948
    76 INDEX=0                                                             MTRS0949
       CALL NPNRMX  (H1,H1,N,FL(13),INDEX,MAXR,NC,2)                       MTRS0950
       CALL NPNRMX  (H2,H2,N,FL(13),INDEX,MAXR,NC,2)                       MTRS0951
                                                                           MTRS0952
       DD  78  J=1,NC                                                      MTRS0953
         J1=2*MAXR*(J-1)                                                   MTRS0954
         DD  78  I=1,NX ,2                                                 MTRS0955
           K=J1+I                                                          MTRS0956
    78     H2(K)=H1(K)-H2(K)                                               MTRS0957
       INDEX1=0                                                            MTRS0958
       CALL DPFDRM (0,H2,H2,1,N,NC,MAXR,MAXR)                              MTRS0959
       CALL NPNRMX  (H2,H2,N,FL(13),INDEX1,MAXR,NC,1)                      MTRS0960
                                                                           MTRS0961
       EIG1=FL(9)                                                          MTRS0962
       EIG2=FL(5)                                                          MTRS0963
       GOTD  (80,79),NC                                                    MTRS0964
    79 EIG1(2)=FL(11)                                                      MTRS0965
       EIG2(2)=FL(7)                                                       MTRS0966
                                                                           MTRS0967
    80 CALL DPFORM (0,U,U,1,N,MX,MAXR,MAXR)                                MTRS0968
       CALL DPFORM  (0,H1,H1,1,N,NC,MAXR,MAXR)                             MTRS0969
       IF ACCUMULATOR OVERFLOW       92,90                                 MTRS0970
    90 IF QUDTIENT DVERFLDW        92,91                                   MTRS0971
```

```
 91 IF DIVIDE CHECK           92,81                                   MTRS0972
 92    IR=-1                                                          MTRS0973
                                                                      MTRS0974
 81 RETURN                                                            MTRS0975
 82 CALL DPFORM (0,GUESS,GUESS,1,N,NC,MAXR,MAXR)                      MTRS0976
    CALL DPFORM  (0,U,U,1,N,MX,MAXR,MAXR)                             MTRS0977
    INDEX1=INDEX                                                      MTRS0978
    EIG1=FL(1)                                                        MTRS0979
    IR=1                                                              MTRS0980
    GOTO  (81,83),NC                                                  MTRS0981
 83 EIG1(2)=FL(3)                                                     MTRS0982
    GOTO  81                                                          MTRS0983
 84 IR=-1                                                             MTRS0984
    GOTO  80                                                          MTRS0985
                                                                      MTRS0986
 85 X=FL(17) / TEST                                                   MTRS0987
    X1=FL(18) / TEST                                                  MTRS0988
    GOTO   (94,97),NC                                                 MTRS0989
 94 X2 = (GUESS(1)-H2(1))**2                                          MTRS0990
    DO  96  I=3,NX,2                                                  MTRS0991
      IF  (X2-(GUESS(I)-H2(I))**2)   95,96,96                         MTRS0992
 95   X2= (GUESS(I)-H2(I))**2                                         MTRS0993
 96   CONTINUE                                                        MTRS0994
    GOTO  100                                                         MTRS0995
 97 X2=(GUESS(1)-H2(1))**2 + (GUESS(2*MAXR+1)-H2(2*MAXR+1))**2        MTRS0996
    DO  99  I=3,NX,2                                                  MTRS0997
      J=MAXR+I                                                        MTRS0998
      IF  (X2 - ( (GUESS(I)-H2(I))**2+(GUESS(J)-H2(J))**2 ) ) 98,99,99MTRS0999
 98   X2= (GUESS(I)-H2(I))**2 + (GUESS(J)-H2(J))**2                   MTRS1000
 99   CONTINUE                                                        MTRS1001
100 X2=SQRTF(X2) / TESTSR                                             MTRS1002
    ITERS=ITERS+1                                                     MTRS1003
    IF (X-X1)                86,87,87                                 MTRS1004
 86 X=X1                                                              MTRS1005
 87 IF (X-X2)                60,60,82                                 MTRS1006
                                                                      MTRS1007
    END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NDT USEO BY PROGRAM

OEC	OCT	OEC	OCT
4956	11534	32561	77461

STORAGE LDCATIDNS FOR VARIABLES APPEARING IN OIMENSION AND EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
P1I	4945	11521	P1	4947	11523	PI	4953	11531	P	4955	11533
Q1I	4941	11515	Q1	4943	11517	QI	4949	11525	Q	4951	11527
X1	4937	11511	X2	4935	11507	X3	4933	11505	X4	4931	11503
X5	4929	11501	X6	4927	11477	X7	4925	11475	X8	4923	11473
X9	4921	11471	X	4939	11513						

STORAGE LDCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, DR EQUIVALENCE STATEMENTS

	DEC	OCT		OEC	OCT		DEC	OCT		OEC	OCT
I1	4919	11467	I2	4918	11466	I3	4917	11465	I4	4916	11464
IGOTO	4915	11463	I	4914	11462	J1	4913	11461	J	4912	11460
K1	4911	11457	K	4910	11456	L	4909	11455	MX	4908	11454
NAK	4907	11453	NX	4906	11452						

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		OEC	OCT
1)	4859	11373	2)	4812	11314	3)	4819	11323	4)	32767	77777
6)	4832	11340	9)	4838	11346	A)GO	4803	11303	C)GO	4887	11427
C)G1	4888	11430	C)G2	4889	11431	C)G4	4890	11432	C)G5	4891	11433
C)G6	4892	11434	C)G7	4893	11435	C)G8	4894	11436	C)G9	4895	11437
C)GA	4896	11440	C)GB	4897	11441	C)GC	4898	11442	C)GD	4899	11443
C)200	4900	11444	C)201	4901	11445	C)205	4902	11446	C)209	4903	11447
C)20B	4904	11450	C)20E	4905	11451	0)118	1783	03367	D)11T	3448	06570
0)13L	4784	11260	D)208	1499	02733	0)20I	1612	03114	D)20J	1616	03120
0)219	1790	03376	D)222	3487	06537	0)226	3527	06707	0)227	3533	06715
D)228	3541	06725	D)237	4644	11044	D)23A	4689	11121	0)308	1498	02732
0)322	3486	06636	D)326	3526	06706	0)327	3532	06714	D)328	3540	06724
D)33A	4688	11120	0)403	1199	02257	0)40C	1570	03042	D)410	3411	06523
D)42E	3637	07065	D)42T	4526	10656	D)432	4599	10767	D)436	4639	11037

D)43G	4726	11166	0)510	3410	06522	0)62E	3636	07064	D)62T	4525	10655
0)636	4638	11036	0)72E	3635	07063	E)E	1598	03076	E)G	1606	03106
E)0	1638	03146	E)P	1642	03152	E)R	1658	03172	E)33	4628	11024
E)3I	4765	11235	E)3K	4782	11256						

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		OEC	OCT		OEC	OCT		DEC	OCT
DEXP(2	3	00003	DPFORM	0	00000	DPMLTX	1	00001	DSQRT	9	00011
NPNRMX	2	00002	SQRT	7	00007	(OFAD)	8	00010	(DFOP)	6	00006
(DFMP)	4	00004	(OFSB)	5	00005						

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

DEXP(2	DPFORM	DPMLTX	DSQRT	NPNRMX	SQRT	(OFAO)	(DFOP)
(DFMP)	(OFSB)						

EXTERNAL FORMULA NUMBERS WITH CORRESPONOING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
5	7	02167	6	8	02171	8	19	02260	9	28	02350
10	29	02352	11	30	02356	12	31	02360	13	39	02734
22	42	03022	14	43	03026	15	44	03033	16	49	03103
17	51	03115	18	52	03121	19	53	03123	190	54	03131
191	56	03141	20	57	03156	21	58	03176	24	62	03237
25	64	03250	28	65	03312	29	66	03314	30	69	03327
31	73	03357	32	74	03370	33	75	03377	35	76	03404
40	98	05661	400	99	05667	401	103	05725	37	104	05744
41	105	05765	42	110	06041	43	113	06062	44	118	06510
45	120	06514	46	124	06552	49	126	06563	50	130	06617
51	132	06630	52	137	06673	54	139	06710	56	140	06716
58	147	06766	59	152	07016	60	158	07066	64	159	07070
65	163	07234	66	166	07264	68	168	07323	70	170	07362
73	173	07503	75	176	07626	77	178	07750	71	183	10153
72	189	10233	74	195	10527	76	196	10606	78	205	10704
79	214	10763	80	216	10770	90	222	11025	91	224	11030
92	225	11033	81	226	11040	82	228	11045	83	236	11112

```
  84   238 11115        85   240 11122        94   243 11133        95   246 11155
  96   247 11163        97   249 11167        98   253 11236        99   254 11253
 100   255 11261        86   258 11274        87   259 11276
```

```
C     SUBROUTINE PICKSUP MATRIX IN SINGLE PRECISION FORMAT AND STORES IT   MTRS1010
C           IN DOUBLE PRECISION FORMAT WITH L.S.H. = ZERO.                 MTRS1011
C          OR.......                                                       MTRS1012
C          PICKS UP DOUBLE PRECISION MATRIX AND STORES IT IN SINGLE        MTRS1013
C          PRECISION FORMAT WITH L.S.H. LOST.                              MTRS1014
C                                                                          MTRS1015
C        NUTAPE = ZERO, MATRIX IS IN CORE                                  MTRS1016
C               = NON-ZERO, MATRIX IS ON TAPE UNIT NUTAPE, BY              MTRS1017
C                 COLUMNS WITH EACH COLUMN = 1 RECORD                      MTRS1018
C           A   = SINGLE PRECISION MATRIX                                  MTRS1019
C           B   = DOUBLE PRECISION MATRIX     (MAY = A)                    MTRS1020
C        ICHUZ  = 0, CONVERT SINGLE PR. TO DOUBLE PR.                      MTRS1021
C               = NON-ZERO, CONVERT DOUBLE PR. TO SINGLE PR.               MTRS1022
C           M   = NUMBER ROWS                                              MTRS1023
C           N   = NUMBER COLUMNS                                           MTRS1024
C          MA   = DIMENSIONED NUMBER ROWS OF A                             MTRS1025
C          MB   = DIMENSIONED NUMBER ROWS OF B                             MTRS1026
C          MA,MB ARE SINGLE PRECISION DIMENSIONS, HOWEVER B MUST BE        MTRS1027
C          LARGE ENUF TO ACCOMODATE THE DOUBLE MATRIX.                     MTRS1028
C               IF MATRIX IS COMPLEX, N=TWICE THE ACTUAL NUMBER OF         MTRS1029
C               COLUMNS. IN THIS CASETHE RESERVED NUMBER OF CORES FOR     MTRS1030
C               A MUST BE AT LEAST 2*M*N(ACTUAL) AND FOR B, AT LEAST       MTRS1031
C               4*M*N(ACTUAL).                                             MTRS1032
C                                                                          MTRS1033
C                                                                          MTRS1034
                                                                           MTRS1035
                                                                           MTRS1036
                                                                           MTRS1037
      SUBROUTINE DPFORM (NUTAPE,A,B,ICHUZ,M,N,MA,MB)                       MTRS1038
                                                                           MTRS1039
      DIMENSION   A(1),B(1)                                                MTRS1040
                                                                           MTRS1041
      MB1 = 2*MB                                                           MTRS1042
      IF ( ICHUZ )   20,10,20                                              MTRS1043
   10 IF ( NUTAPE )  11,13,11                                              MTRS1044
                                                                           MTRS1045
   11 REWIND NUTAPE                                                        MTRS1046
```

```
      K = 1                                                             MTRS1047
      DO  12  I=1,N                                                     MTRS1048
        KK = K+M                                                        MTRS1049
        READ TAPE NUTAPE,  (A(J),J=K,KK)                                MTRS1050
12      K = K+M                                                         MTRS1051
      MB1 = 2*M                                                         MTRS1052
      GOTO  14                                                          MTRS1053
                                                                        MTRS1054
13 IX = 1                                                               MTRS1055
      IF (M-MB)  15,14,15                                               MTRS1056
14 IX=1                                                                 MTRS1057
                                                                        MTRS1058
15 DO  16  I=1,N                                                        MTRS1059
        K = (N-I)*MA+M+1                                                MTRS1060
        KK = (N-I)*MB1*IX+2*M+1                                         MTRS1061
        DO  16  J=1,M                                                   MTRS1062
          IGET = K-J                                                    MTRS1063
          IPUT = KK-2*J                                                 MTRS1064
          B(IPUT) = A(IGET)                                             MTRS1065
16        B(IPUT+1) = 0.                                                MTRS1066
      GOTO  26                                                          MTRS1067
                                                                        MTRS1068
20 IF (NUTAPE)  21,23,21                                                MTRS1069
21 REWIND NUTAPE                                                        MTRS1070
      K = 1                                                             MTRS1071
      DO  22  I=1,N                                                     MTRS1072
        KK = K+2*M                                                      MTRS1073
        READ TAPE NUTAPE,  (B(J),J=K,KK)                                MTRS1074
22      K = K+2*M                                                       MTRS1075
      MB1 = 2*M                                                         MTRS1076
                                                                        MTRS1077
23 K = 0                                                                MTRS1078
      KK = 0                                                            MTRS1079
      DO  25  I=1,N                                                     MTRS1080
        DO  24  J=1,M                                                   MTRS1081
          IGET = KK+2*J-1                                               MTRS1082
          IPUT = K+J                                                    MTRS1083
24        A(IPUT) = B(IGET)                                             MTRS1084
```

```
      K = K+MA                                                          MTRS1085
25    KK = KK+MB1                                                       MTRS1086
                                                                        MTRS1087
26 RETURN                                                               MTRS1088
   END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
280	00430	32561	77461

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
IGET	279	00427	IPUT	278	00426	I	277	00425	IX	276	00424
J	275	00423	KK	274	00422	K	273	00421	MB1	272	00420

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
1)	265	00411	2)	253	00375	3)	256	00400	6)	257	00401
9)	263	00407	C)G1	268	00414	C)G2	269	00415	C)202	270	00416
C)205	271	00417	0)20A	115	00163	0)20L	215	00327	0)30A	114	00162
0)30L	214	00326									

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
(RLR)	2	00002	(RWT)	0	00000	(TSB)	1	00001			

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

(RLR) (RWT) (TSB)

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
10	6	00103	11	7	00105	12	16	00131	13	19	00143
14	21	00152	15	22	00154	16	29	00240	20	31	00253
21	32	00255	22	41	00303	23	43	00316	24	49	00352
25	51	00362	26	52	00371						

```
C      SUBROUTINE                                                      MTRS0035
C                                                                      MTRS0036
C  CALLING SEQUENCE.....                                               MTRS0037
C     CALL DPMLTX (A,NA,B,NB,C,M,N,K,MA,MB,MC,NP)                      MTRS0038
C                                                                      MTRS0039
C          A = PREMULTIPLIER         MA = DIMENSIONED NUMBER OF ROWS A MTRS0040
C          B = POSTMULTIPLIER        MB = DIMENSIONED NUMBER OF ROWS B MTRS0041
C          C = PRODUCT               MC = DIMENSIONED NUMBER OF ROWS C MTRS0042
C          M = NO. ROWS IN A     = NO. ROWS IN C                       MTRS0043
C          N = NO. COLUMNS IN A = NO. ROWS IN B                        MTRS0044
C          K = NO. COLUMNS IN B = NO. COLUMNS IN C                     MTRS0045
C          NA AND NB = 1 IF A OR B, RESPECTIVELY, ARE REAL.            MTRS0046
C                    = 2 IF A OR B, RESPECTIVELY, ARE COMPLEX.         MTRS0047
C          NP= PRECISION,  (1 OR 2)                                    MTRS0048
C          MA,MB,MC ARE SINGLE PRECISION DIMENSIONS,(1/2 OF ACTUAL CORE MTRS0049
C           RESERVED FOR REAL DOUBLE PRECISION MATRICES, AND 1/4 OF    MTRS0050
C           ACTUAL CORE RESERVED FOR COMPLEX DOUBLE PRECISON MATRICES) MTRS0051
                                                                       MTRS0052
      SUBROUTINE DPMLTX (A,NA,B,NB,C,M,N,K,MA,MB,MC,NP)                MTRS0053
                                                                       MTRS0054
D     DIMENSION  A(1), B(1), C(I)                                      MTRS0055
                                                                       MTRS0056
      IA=MC*K*NP                                                       MTRS0057
      IB=MA*N*NP*NA                                                    MTRS0058
      IC=1                                                             MTRS0059
      ID=MA*NP*NA                                                      MTRS0060
      IH=MC*NP                                                         MTRS0061
      IJ=MC*NP                                                         MTRS0062
      IK=M*NP                                                          MTRS0063
      IL=1*NP                                                          MTRS0064
      IM=1-NP                                                          MTRS0065
      GOTO  (6,7),NA                                                   MTRS0066
    6 GOTO  (11,10),NB                                                 MTRS0067
    7 GOTO  (9,8),NB                                                   MTRS0068
    8       IC=2                                                       MTRS0069
            GOTO  10                                                   MTRS0070
    9 IH=2*IH                                                          MTRS0071
```

```
      IC=3                                                              MTRS0072
   10 IA=2*IA                                                           MTRS0073
      IJ=2*IJ                                                           MTRS0074
   11 DO  30  I=1,IK,IL                                                 MTRS0075
      INC=IM                                                            MTRS0076
        DO  16  J=I,IA,IH                                               MTRS0077
      IN=INC                                                            MTRS0078
          GOTO (12,14),NP                                               MTRS0079
                                                                        MTRS0080
   12   C(J)=0.                                                         MTRS0081
        DO  13  L=I,IB,ID                                               MTRS0082
          IN=IN+NP                                                      MTRS0083
   13     C(J)=C(J)+A(L)*B(IN)                                          MTRS0084
          GOTO  16                                                      MTRS0085
                                                                        MTRS0086
D  14                   C(J)=0.                                         MTRS0087
                        DO  15  L=I,IB,ID                               MTRS0088
                          IN=IN+NP                                      MTRS0089
D  15                     C(J)=C(J)+A(L)*B(IN)                          MTRS0090
   16   INC=INC+NP*MB                                                   MTRS0091
                                                                        MTRS0092
      GOTO  (30,18,24),IC                                               MTRS0093
   18 IE=1+NP*MA                                                        MTRS0094
      INC=IM                                                            MTRS0095
      DO  23  J=I,IA,IJ                                                 MTRS0096
      IN=INC                                                            MTRS0097
        IF=J+NP*MC                                                      MTRS0098
      GOTO (19,21),NP                                                   MTRS0099
   19   DO  20  L=IE,IB,ID                                              MTRS0100
          IN=IN+1                                                       MTRS0101
          IG=IN+MB                                                      MTRS0102
          C(IF)=C(IF)+A(L)*B(IN)                                        MTRS0103
   20     C(J)=C(J)-A(L)*B(IG)                                          MTRS0104
          GOTO 23                                                       MTRS0105
   21   DO  22  L=IE,IB,ID                                              MTRS0106
          IN=IN+NP                                                      MTRS0107
          IG=IN+NP*MB                                                   MTRS0108
D         C(IF)=C(IF)+A(L)*B(IN)                                        MTRS0109
```

```
D  22      C(J)=C(J)-A(L)*B(IG)                                MTRS0110
                                                               MTRS0111
   23    INC=INC+2*NP*MB                                       MTRS0112
         GOTO  30                                              MTRS0113
   24    IE=I+NP*MC                                            MTRS0114
         IF=I+NP*MA                                            MTRS0115
       INC=IM                                                  MTRS0116
         DO  29  J=IE,IA,IJ                                    MTRS0117
       IN=INC                                                  MTRS0118
           GOTO  (25,27),NP                                    MTRS0119
   25    C(J)=0.                                               MTRS0120
         DO  26  L=IF,IB,ID                                    MTRS0121
           IN=IN+1                                             MTRS0122
   26      C(J)=C(J)+A(L)*B(IN)                                MTRS0123
         GOTO  29                                              MTRS0124
D  27    C(J)=0.                                               MTRS0125
         DO  28  L=IF,IB,ID                                    MTRS0126
           IN=IN+NP                                            MTRS0127
D  28      C(J)=C(J)+A(L)*B(IN)                                MTRS0128
   29    INC=INC+NP*MB                                         MTRS0129
                                                               MTRS0130
   30    CONTINUE                                              MTRS0131
       RETURN                                                  MTRS0132
       END(1,0,0,0,0,0,0,0,0,1,0,0,0,0,0)
```

STORAGE NOT USED BY PROGRAM

DEC	OCT	DEC	OCT
606	01136	32561	77461

STORAGE LOCATIONS FOR VARIABLES NOT APPEARING IN COMMON, DIMENSION, OR EQUIVALENCE STATEMENTS

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
IA	605	01135	IB	604	01134	IC	603	01133	ID	602	01132
IE	601	01131	IF	600	01130	IG	599	01127	IH	598	01126
IJ	597	01125	IK	596	01124	IL	595	01123	IM	594	01122
INC	593	01121	IN	592	01120	I	591	01117	J	590	01116

LOCATIONS FOR OTHER SYMBOLS NOT APPEARING IN SOURCE PROGRAM

	DEC	OCT		DEC	OCT		DEC	OCT		DEC	OCT
2)	566	01066	3)	569	01071	4)	32767	77777	6)	572	01074
9)	578	01102	C)G0	580	01104	C)G1	581	01105	C)G2	582	01106
C)G3	583	01107	C)G4	584	01110	C)G5	585	01111	C)G6	586	01112
C)G7	537	01113	C)G8	588	01114	C)200	589	01115	D)107	242	00362
D)115	556	01054	D)40I	343	00527	E)1	206	00316			

LOCATIONS OF NAMES IN TRANSFER VECTOR

	DEC	OCT		DEC	OCT	DEC	OCT	DEC	OCT
(DFAD)	1	00001	(DFMP)	0	00000				

ENTRY POINTS TO SUBROUTINES NOT OUTPUT FROM LIBRARY

(DFAD) (DFMP)

EXTERNAL FORMULA NUMBERS WITH CORRESPONDING INTERNAL FORMULA NUMBERS AND OCTAL LOCATIONS

EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC	EFN	IFN	LOC
6	14	00317	7	15	00322	8	16	00325	9	18	00332
10	20	00342	11	22	00352	12	27	00401	13	30	00416
14	32	00425	15	35	00447	16	36	00473	18	38	00510

19	44 00546	20	48 00573	21	50 00604	22	54 00650
23	55 00701	24	57 00716	25	63 00751	26	66 00766
27	68 00775	28	71 01017	29	72 01043	30	73 01055

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilizes a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilizes a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilizes a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilized a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilizes a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilizes a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilizes a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

Aerospace Corporation, El Segundo, California. FLUTTER AND VIBRATION ANALYSIS BY A MODAL METHOD: ANALYTICAL DEVELOPMENT AND COMPUTATIONAL PROCEDURE, prepared by W. P. Rodden, E. F. Farkas, and H. A. Malcom. 31 July 1963. [221]p. incl. illus. (Report TDR-169(3230-11)TN-15;SSD-TDR-63-158) (Contract AF 04(695)-169) Unclassified report

A modal solution of the flutter and vibration problems for a multiple component system is presented. The formulation requires the aerodynamic characteristics in the form of aerodynamic influence coefficients, and the structural and inertial characteristics in the form of free vibration modes and frequencies and a mass matrix for each component. The use of a rigid-body modal matrix permits a general analysis for a system free in space with up to six rigid-body degrees of freedom. The solution also utilized a newly developed

(over)

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED

UNCLASSIFIED

subroutine for finding the eigenvalues and eigenvectors of a non-Hermitian matrix having close roots, and obtains the corresponding values of frequency, required structural damping, and velocity of the flutter system. The Aerospace IBM 7090 Computer Program No. LD014A provides the flutter or vibration solution for a system composed of as many as 20 flexible components, each of which may have up to 50 control points. The solution is carried out by a modal (series) method choosing a series of up to 40 cantilever and/or free-free vibration modes to approximate the control point deflections. The program may be used to find the vibration characteristics of a composite system if the cantilever vibration characteristics of the components are known, and the generalized masses may be obtained for use in subsequent modal analyses of flying qualities or stability and control of flexible vehicles. An option is provided to vary the density as well as the reduced velocity in the flutter analysis.

UNCLASSIFIED